



**Universidade de
Aveiro
2008**

Departamento de Matemática

**Mónica Ferreira de
Jesus**

**Obtenção de Limites Inferiores para Problemas de
Desenho de Redes**



**Universidade de
Aveiro
2008**

Departamento de Matemática

**Mónica Ferreira de
Jesus**

**Obtenção de Limites Inferiores para Problemas de
Desenho de Redes**

Dissertação apresentada à Universidade de Aveiro para cumprimento dos requisitos necessários à obtenção do grau de Mestre em Matemática, realizada sob a orientação científica da Doutora Maria Cristina Saraiva Requejo Agra, Professora Auxiliar do Departamento de Matemática da Universidade de Aveiro.

o júri

presidente

Professor Doutor Domingos Moreira Cardoso
professor Catedrático da Universidade de Aveiro

Professora Doutora Maria Adelaide da Cruz Cerveira
professora Auxiliar da Universidade de Trás-Os-Montes e Alto Douro

Professora Doutora Maria Cristina Saraiva Requejo Agra
professora Auxiliar da Universidade de Aveiro (Orientadora)

agradecimentos

À Professora Doutora Maria Cristina Saraiva Requejo Agra, orientadora da dissertação, agradeço o apoio e todas as contribuições para o trabalho. Agradeço a todos os meus familiares e amigos, em especial aos meus pais, pelo incentivo, pela paciência e atenção.

palavras-chave

Relaxação lagrangeana, Problema da árvore de suporte de custo mínimo com restrições de salto, Método subgradiente, Método subgradiente radar, Método do feixe de subgradientes.

resumo

Nesta tese descrevemos a técnica da relaxação lagrangeana e métodos para otimizar o problema dual lagrangeano. Descrevemos o método do subgradiente, o método do subgradiente radar e o método do feixe de subgradientes. Em particular apresentamos uma relaxação lagrangeana para o problema da Árvore de Suporte de Custo Mínimo com Restrições de Salto (HMST) e aplicamos alguns destes métodos à sua resolução. O problema HMST surge, por exemplo, em situações de desenho de redes de telecomunicações centralizadas com restrições de qualidade de serviço e consiste em determinar uma árvore de suporte de custo mínimo com a restrição adicional de o número de arestas no único caminho de um nodo central a qualquer outro nodo da árvore não ser superior a um determinado valor. Apresentamos, ainda alguns resultados computacionais obtidos com a aplicação de alguns dos métodos descritos para a obtenção de uma aproximação do valor da relaxação lagrangeana apresentada para o problema HMST.

keywords

Lagrangian relaxation, Hop-constrained minimum spanning tree problem, Subgradient method, Radar subgradient method, Bundle method.

abstract

In this thesis we describe the lagrangean relaxation technique and methods to optimize the lagrangean dual problem. We describe the subgradient method, the radar subgradient method and the bundle method. We present a lagrangean relaxation for the Hop-constrained minimum spanning tree problem (HMST) and use some of the described methods to solve this relaxation. The HMST appears in centralized network design problems where quality of service constraints are required and the problem is to determine a minimum spanning tree such that the length of the only path between the central node and any node in the tree is at most a certain value. We also present some computational results obtained when solving the lagrangean relaxation presented for the HMST using some the methods described.

Conteúdo

1	Introdução	1
2	Relaxação Lagrangeana	7
3	Métodos de resolução da relaxação lagrangeana	19
3.1	Método subgradiente	21
3.2	Método subgradiente radar	25
3.3	Método do feixe de subgradientes	28
4	O problema HMST e uma sua relaxação lagrangeana	35
4.1	O problema HMST	35
4.2	Formulação de fluxos orientada com índices de salto	37
4.3	Uma relaxação lagrangeana	40
4.4	Aplicação dos métodos ao problema	43
5	Resultados computacionais	47
5.1	Resultados obtidos pelo método subgradiente	50
5.2	Resultados obtidos pelo método subgradiente radar	59
5.3	Comparação dos resultados obtidos entre os dois métodos	65
6	Conclusão	73

Capítulo 1

Introdução

O objectivo principal deste trabalho consiste em descrever métodos [1, 3, 4, 6, 16] para otimizar o problema dual lagrangeano [9, 12]. Em particular, descreveremos uma relaxação lagrangeana para o problema da Árvore de Suporte de Custo Mínimo com Restrições de Salto [15], que designaremos por HMST (*Hop-constrained Minimum Spanning Tree*), e aplicaremos alguns dos métodos descritos a este problema.

A relaxação lagrangeana é uma técnica muito usada na obtenção de limites (inferiores) para muitos problemas de Optimização Combinatória. Muitas vezes várias relaxações lagrangeanas são possíveis para o mesmo problema dependendo do conjunto de restrições relaxadas e possuindo propriedades distintas embora algumas com o mesmo limite (inferior) teórico.

No problema HMST pretende-se determinar uma árvore de suporte de custo mínimo com a restrição adicional de o número de arestas no único caminho desde o nodo raíz até qualquer outro nodo da árvore não ser superior a um determinado valor H . Este problema ocorre em situações de desenho de redes de telecomunicações centralizadas onde restrições adicionais, neste caso restrições de salto, estão relacionadas com a *performance* da rede.

Uma rede de telecomunicações pode ser definida como um sistema para a troca de informações à distância [18]. É constituída por nodos e arcos que possibilitam a interconexão entre diferentes pontos de acesso à rede. Nodo é todo o ponto de uma rede onde se tem comutação de sinais e arco é o meio de transmissão que une dois nodos.

No problema HMST consideramos um nodo raíz que corresponde ao computador central e os restantes nodos correspondem a terminais que têm de estar ligados ao computador central e as arestas correspondem aos potenciais fios de interconexão.

A topologia de uma rede refere-se ao modo como os elementos da rede são interligados. As topologias básicas utilizadas são em barra, em árvore, em estrela, em malha e em anel (ver Figura 1.1) [18].



Figura 1.1: Topologias básicas de uma rede de telecomunicações

Numa rede em barra todos os dispositivos estão ligados directamente à linha por onde circulam os pacotes, pelo que todos os dispositivos da rede vêem os pacotes. Cada dispositivo da rede tem um endereço único, que permite através da análise dos pacotes seleccionar os que lhe são destinados.

Existe uma forma ligeiramente mais complexa desta topologia, a denominada topologia em árvore, cuja barra começa num ponto denominado raíz e após esse ponto partem vários ramos que têm ligados os dispositivos que compõem a rede. O ponto de onde saem os ramos é obtido por um simples conector.

Como o nome indica, a topologia em estrela tem a forma de uma estrela e consiste em vários cabos que unem cada dispositivo a um ponto central. Para que uma rede tenha topologia em estrela não é necessário ter uma disposição em forma de estrela, é necessário somente cada dispositivo da rede estar ligado por um cabo próprio a um ponto central.

A topologia em malha caracteriza-se por usar uma transmissão ponto-a-ponto, ou seja, cada par de componentes liga-se e comunica directamente. Normalmente usada em redes de longa distância, é a topologia mais dispendiosa e mais complexa de entre todas visto requerer muitas ligações.

Na topologia em anel os pacotes circulam por todos os dispositivos da rede, tendo cada um o seu endereço. O fluxo de informação é unidireccional, existindo um dispositivo que intercepta e gere o fluxo de dados que entra e sai do anel.

Para além de ser necessário descrever a topologia da rede, essa topologia pode ter de satisfazer algum tipo de restrições adicionais. Um tipo de restrições adicionais que pode ser usada e que são consideradas no problema HMST são as restrições de salto. As restrições de salto limitam o número de ligações que podem ser utilizadas para ligar o computador central a qualquer um dos terminais e garantem uma certa qualidade de serviço no que diz respeito a algumas restrições de *performance*, tais como, a disponibilidade (probabilidade de todas as linhas de transmissão no caminho do terminal ao computador central estarem a funcionar), a fiabilidade (probabilidade de uma sessão não ser interrompida por falha de uma ligação) e os tempos de atraso máximo de transmissão. Assim, caminhos com menos saltos originam redes com um melhor desempenho visto que caminhos com muitos nodos podem originar tempos de transmissão muito elevados. Deste modo, atribuindo um valor baixo para o número máximo de saltos permitidos diminui a probabilidade de falha em cada caminho e as restrições evitam a degradação da qualidade do sinal.

Como muitos outros problemas de Optimização Combinatória, o problema HMST é \mathcal{NP} -difícil pelo que não nos é possível garantir a sua resolução até à optimalidade em tempo polinomial e por este motivo, é necessário desenvolver métodos que determinem bons limites inferiores (e/ou superiores). Para determinarmos o valor óptimo ou valores aproximados de um problema \mathcal{NP} -difícil podemos usar vários tipos de processos. Um dos processos consiste no uso de relaxações lagrangeanas, que permitem obter o valor óptimo ou valores aproximados do problema recorrendo a uma formulação dual lagrangeana que origina uma relaxação do problema [12] e, tal como já referimos, será o que aplicaremos neste trabalho.

O problema HMST é \mathcal{NP} -difícil porque contém, como caso particular (o caso em que $H = 2$), uma versão \mathcal{NP} -difícil do problema de Localização Simples Sem Restrições de Capacidade (*Simple Uncapacitated Facility Location* [15]) onde os locais de potencial localização de facilidades coincidem com a localização de clientes a serem servidos.

Neste trabalho descrevemos alguns métodos que permitam obter aproximações para o valor óptimo do problema HMST obtendo-se limites inferiores para o valor da solução óptima da relaxação lagrangeana deste problema de optimização combinatória. Descrevemos os métodos Subgradiente, Subgradiente Radar e do Feixe de Subgradientes e aplicá-los-emos, sempre que possível, à resolução da relaxação lagrangeana do problema HMST apresentada. Embora várias relaxações lagrangeanas sejam possíveis para o problema HMST, dependendo do conjunto de restrições relaxadas, neste trabalho apresentamos apenas uma relaxação lagrangeana, que se mostrou superior às restantes [15]. Os métodos de optimização do problema dual lagrangeano são métodos genéricos que são usados para resolver problemas de optimização não diferenciável e que em particular são muito usados na obtenção de uma aproximação do valor do problema dual lagrangeano [4].

O método subgradiente é muito idêntico ao método gradiente para funções diferenciáveis, mas com algumas excepções. Por exemplo, o método subgradiente usa tamanhos de passo que são prefixados, em vez de uma linha de procura (aproximada ou exacta). O método subgradiente não é um método de descida como o método gradiente visto que o valor da função pode (e costuma) aumentar ou diminuir, provocando um comportamento zig-zag. O método subgradiente pode ser algo demorado a atingir o melhor valor possível mas é simples e pode ser aplicado a uma grande variedade de problemas. Tal como acontece no método subgradiente, no método subgradiente radar usamos a direcção do subgradiente e a aproximação de primeira ordem da função dual para determinar o tamanho do passo na direcção subgradiente. A diferença entre o método subgradiente clássico e o método subgradiente radar consiste na forma como o próximo ponto é obtido uma vez que usamos uma aproximação de primeira ordem da função dual que é construída usando todo o conjunto de subgradientes já calculados. O método do feixe de subgradientes tem como objectivo usar toda a informação determinada ao longo das sucessivas iterações, em particular usa também, mas de modo diferente o conjunto de subgradientes já calculados e tem como finalidade forçar o aumento do valor da função dual. Este método melhora os métodos já referidos porque tem a capacidade de prevenir instabilidade e possui técnicas de agregação para controlar

o tamanho do feixe.

Esta dissertação está organizada do seguinte modo. No Capítulo 2 fazemos uma descrição da técnica da relaxação lagrangeana de um problema de programação linear.

No Capítulo 3 apresentamos três métodos de resolução da relaxação lagrangeana, que são métodos genéricos usados para resolver problemas de optimização não diferenciável. Primeiro, abordamos o método subgradiente cujo algoritmo é bastante simples para a minimização de funções não-diferenciáveis. Descrevemos, de seguida, o método subgradiente radar que foi desenvolvido para usar uma aproximação de primeira ordem da função dual dada pelo conjunto de subgradientes já calculados. Por último, fazemos uma breve abordagem ao método do feixe de subgradientes cuja implementação não foi possível devido à extrema complexidade do algoritmo, visto que este método inclui resolução de problemas quadráticos.

No Capítulo 4 descrevemos o problema HMST, uma formulação de fluxos orientada com índices de salto (a formulação HMCF) e uma sua relaxação lagrangeana.

No Capítulo 5 apresentamos os resultados computacionais, descrevemos as instâncias usadas, fornecemos os resultados obtidos pelos métodos subgradiente e subgradiente radar e as respectivas comparações efectuadas.

Por fim, no Capítulo 6 são feitos alguns comentários ao trabalho desenvolvido e algumas considerações finais.

Capítulo 2

Relaxação Lagrangeana

A técnica da relaxação lagrangeana [9, 12, 19] é baseada na ideia de não considerarmos algumas restrições da formulação original tentando, desse modo, obter problemas de mais fácil resolução que permitam determinar bons limites (inferiores) para o valor óptimo do problema. Com este propósito, para determinar o valor óptimo ou valores aproximados para o valor do problema podemos considerar a construção de relaxações lagrangeanas onde se relaxam as restrições que complicam o modelo.

Neste capítulo fazemos uma breve descrição da técnica da relaxação lagrangeana e para tal consideramos um problema de programação linear inteira (P) definido da seguinte forma:

$$\begin{aligned} (P) \quad & \min \quad cx \\ & \text{sujeito a: } Ax \geq b \\ & \quad \quad Dx \geq f \\ & \quad \quad x \geq 0 \text{ e inteiros} \end{aligned}$$

onde $x \in \mathbb{R}^n$, $c \in \mathbb{R}^n$, $b \in \mathbb{R}^{m_1}$ e $f \in \mathbb{R}^{m_2}$ são vectores, $A \in \mathbb{R}^{m_1 \times n}$ e $D \in \mathbb{R}^{m_2 \times n}$ matrizes e assumimos que se definirmos $X = \{x \in \mathbb{Z}_+^n : Dx \geq f\}$ o conjunto X é compacto (para que isso aconteça basta que as variáveis possuam limite superior e podemos sempre considerar como limite superior valores maiores que os da solução óptima de (P)). Vamos, portanto, considerar que podemos dividir o conjunto de restrições em dois subconjuntos e que o podemos fazer de tal forma que se não se considerarem as restrições $Ax \geq b$ o subproblema resultante será de fácil resolução. Descrevemos

de seguida a construção de uma relaxação lagrangeana que se obtém relaxando estas restrições.

A relaxação lagrangeana de (P) relativamente ao conjunto de restrições $Ax \geq b$ obtém-se associando o vector dual w não negativo, $w \in \mathbb{R}_+^{m_1}$, (o vector dos multiplicadores de Lagrange) ao conjunto de restrições $Ax \geq b$ e construindo o seguinte problema:

$$\theta(w) = \min_{x \in X} \{cx + w(b - Ax)\}.$$

A $\theta(w)$ chamamos função dual lagrangeana de (P) e a w chamamos o vector dos multiplicadores de Lagrange. Este problema não possui as restrições que complicam o modelo pois foram incluídas na função objectivo com uma penalidade de w . Como w é não negativo, se as restrições $Ax \geq b$ forem violadas, a penalidade toma valores positivos e, portanto, as restrições serão satisfeitas se w tomar valores convenientemente grandes.

A função dual lagrangeana de (P) é uma relaxação do problema (P) para qualquer vector $w \geq 0$. Se $\vartheta(P)$ designar o valor óptimo do problema (P) temos o seguinte resultado [12].

Proposição 2.1. $\theta(w) \leq \vartheta(P)$ para todo o $w \geq 0$.

Demonstração. Seja $x^* \in X$ o óptimo de (P) . Esse ponto é forçosamente admissível, pelo que $Ax^* \geq b$. Nestas condições, para qualquer $w \geq 0$ temos $w(b - Ax^*) \leq 0$ e portanto

$$\begin{aligned} \theta(w) &= \min_{x \in X} \{cx + w(b - Ax)\} \\ &\leq cx^* + w(b - Ax^*) \\ &\leq cx^* = \vartheta(P) \end{aligned}$$

□

Relativamente à função dual lagrangeana $\theta(w)$ notamos as seguintes propriedades:

- $\theta(w)$ é uma função contínua para todo $w \geq 0$. Neste caso particular, em que a função objectivo é linear da forma cx , $\theta(w)$ é uma função linear por partes.

- $\theta(w) < +\infty$ para todo $w \geq 0$. Como $\theta(w)$ é uma função contínua e X é compacto, podemos concluir que $\theta(w)$ tem valor finito para todo $w \geq 0$.
- $\theta(w)$ é côncava. Podemos dizer que $\theta(w)$ é uma função côncava se $\theta(w)$ for de minimização e é uma função convexa se $\theta(w)$ for de maximização.
- $\theta(w)$ é diferenciável em quase toda a parte. Mais precisamente, ela não é diferenciável na intersecção dos hiperplanos que a definem (ver Figura 2.1).
- $\theta(w)$ é subdiferenciável para todo $w \geq 0$. (Consideraremos esta propriedade um pouco mais à frente.)

Na Figura 2.1, exemplificamos uma função $\theta(w)$ que é convexa e linear por partes. A figura exhibe uma família de funções lineares para todas as soluções da relaxação lagrangeana que são óptimas para, pelo menos, um valor de w . Considerando a função

$$\theta(w) = \max(16 - 8w)x_1 + (10 - 2w)x_2 + (0 - w)x_3 + (4 - 4w)x_4 + 10w$$

sujeito a:

$$x_1 + x_2 \leq 1$$

$$x_3 + x_4 \leq 1$$

$$x_j \in \{0, 1\}.$$

e para $x_1 = x_2 = x_3 = x_4 = 0$ obtemos $\theta(w) = 10w$,

para $x_2 = 1$ e $x_1 = x_3 = x_4 = 0$ obtemos $\theta(w) = 10 + 8w$,

para $x_2 = x_4 = 1$ e $x_1 = x_3 = 0$ obtemos $\theta(w) = 14 + 4w$,

para $x_1 = 1$ e $x_2 = x_3 = x_4 = 0$ obtemos $\theta(w) = 16 + 2w$,

para $x_1 = x_4 = 1$ e $x_2 = x_3 = 0$ obtemos $\theta(w) = 20 + 2w$.

As restantes condições não estão presentes porque são redundantes para a determinação do envolvente superior da família de equações lineares.

O facto de maximizarmos a função objectivo dual lagrangeana, significa que para qualquer valor de w , $\theta(w)$ é igual ao maior valor destas funções lineares [9]. Portanto, a função $\theta(w)$ é dada pelo conjunto superior desta família de equações lineares por partes que na figura corresponde à parte mais escurecida. A função $\theta(w)$ é convexa e

diferenciável quase em toda a parte, excepto nos pontos onde o problema lagrangeano tem soluções óptimas múltiplas e que correspondem à intersecção das equações lineares.

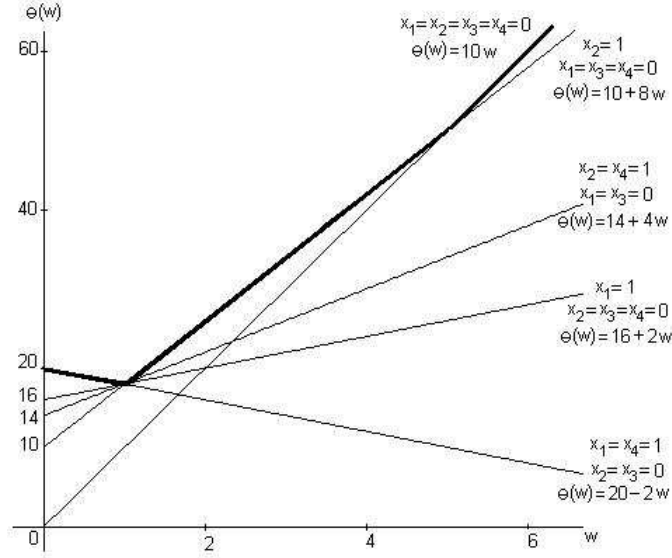


Figura 2.1: A função linear por partes $\theta(w)$

O melhor limite inferior que é possível obter de entre todas as relaxações lagrangeanas $\theta(w)$, com $w \geq 0$, é fornecido através da resolução do problema

$$\theta = \max_{w \geq 0} \theta(w)$$

ao qual chamamos de problema dual lagrangeano [12].

Infelizmente, no caso geral, $\theta(w) \neq \vartheta(P)$. Contudo, como consequência da Proposição 2.1 temos o seguinte resultado.

Proposição 2.2. (*Dualidade Fraca*)

$$\theta \leq \vartheta(P).$$

De um modo geral $\theta < \vartheta(P)$, no entanto, o valor de θ , se bastante próximo de $\vartheta(P)$, é de grande utilidade na resolução do primal utilizando um algoritmo de enumeração implícita. Além disso temos (ver [12]) o seguinte.

Proposição 2.3. *O valor do problema dual lagrangeano satisfaz a desigualdade*

$$\theta \geq \vartheta(P_L)$$

em que P_L designa a relaxação linear do problema (P) que se obtém quando se removem as condições de integrabilidade das variáveis x do problema.

A não diferenciabilidade da função $\theta(w)$ é um inconveniente à resolução do problema dual. Contudo, e uma vez que $\theta(w)$ é côncava, é possível generalizar a noção de subgradiente (ver [4]).

Definição 2.1. Um ponto $y \in \mathbb{R}^{m_1}$ diz-se um subgradiente de θ em $\bar{w} \in \mathbb{R}_+^{m_1}$ se e só se para todo o $w \in \mathbb{R}_+^{m_1}$ se tem

$$\theta(w) - \theta(\bar{w}) \leq y(w - \bar{w}).$$

O conjunto dos subgradientes de θ em w pode ser caracterizado da seguinte forma.

Proposição 2.4. Seja $X(w)$ o conjunto das soluções óptimas de $\theta(w)$, isto é,

$$X(w) = \{x^* \in X : cx^* + w(b - Ax^*) = \min_{x \in X} [cx + w(b - Ax)] = \theta(w)\}$$

e seja

$$Y(w) = \{g^* \in \mathbb{R}_+^{m_1} : g^* = b - Ax^* \text{ e } x^* \in X(w)\}$$

então os elementos de $Y(w)$ são subgradientes de θ em w .

Demonstração. Tomemos $\bar{w} \geq 0$, $\bar{w} \in \mathbb{R}_+^{m_1}$. Temos para todo $x^* \in X(w)$, $\theta(\bar{w}) \leq cx^* + \bar{w}(b - Ax^*)$. Por outro lado, $\theta(w) = cx^* + w(b - Ax^*)$. Subtraindo estas desigualdades obtemos $\theta(\bar{w}) - \theta(w) \leq (b - Ax^*)(\bar{w} - w)$, ou seja, os vectores $g^* = b - Ax^*$ são subgradientes da função $\theta(w)$. \square

Alguns exemplos para a estrutura do conjunto de restrições X e em particular para o conjunto de restrições $Dx \geq f$, que tornam a função dual lagrangeana de fácil resolução, são indicadas a seguir (ver, por exemplo, [12]). Para além de especificarem limites superiores em algumas ou todas as variáveis, adicionalmente terão de ser tais que

- a solução óptima de $\theta(w)$ pode ser obtida por inspecção dos sinais dos coeficientes dos vectores de x , nomeadamente $(c - wA)$;

- incluem algumas restrições limitadas superiormente e que podem ser generalizadas da seguinte forma:

$$\sum_{j \in J_k} x_j = 1, \quad k = 1, 2, \dots, K$$

onde J_1, \dots, J_K são subconjuntos disjuntos de I . A solução ótima de $\theta(w)$ pode ser obtida novamente por inspeção procurando o menor $(c - wA)_j$ estando necessariamente em cada subconjunto J_k ;

- incluem algumas restrições da seguinte forma:

$$\sum_{j \in J_k} \beta_{kj} x_j \leq \beta_{kk} x_k, \quad k = 1, 2, \dots, K$$

onde os k subconjuntos $\{k, J_k\}$ são disjuntos, x_1, \dots, x_K são variáveis 0 - 1, as variáveis em J_k possuem valores contínuos, e todos os coeficientes β são estritamente positivos. Este tipo de restrições surge, por exemplo, em modelos de localização. A relaxação lagrangeana $\theta(w)$ resolve-se facilmente porque separa o problema em K problemas independentes da seguinte forma:

$$\begin{aligned} \min \quad & \sum_{j \in J_k} (c - wA)_j x_j + (c - wA)_k x_k \\ \text{sujeito a:} \quad & \sum_{j \in J_k} \beta_{kj} x_j \leq \beta_{kk} x_k, \quad k=1, \dots, K \quad (1) \\ & 0 \leq x_j \leq u_j, \quad j \in J_k \\ & x_k \in \{0, 1\}, \quad k = 1, \dots, K \end{aligned}$$

onde u_j é um limite superior nas variáveis x_j . Se $x_k = 0$ e da positividade de β_{kj} , tem-se que $x_j = 0$ para todo o $j \in J_k$. Se $x_k = 1$, (1) torna-se um problema trivial de *Knapsack* contínuo com variáveis limitadas. A melhor das soluções obtidas sobre os dois casos $x_k = 0$ e $x_k = 1$ leva à solução ótima de (1). Destas K soluções pode-se determinar a solução ótima de $\theta(w)$.

Estes exemplos de restrições especiais $Dx \geq f$ são os mais simples que podem ser otimizados facilmente. Outras aplicações podem necessitar de algoritmos específicos um pouco mais elaborados.

A potencial utilidade de uma relaxação lagrangeana de (P) é determinada pela proximidade do valor ótimo da relaxação ao valor ótimo de (P) . O maior limite inferior de entre todas as relaxações $\theta(w)$ é $\theta(w^*)$, onde w^* é a solução ótima do problema. Interessa agora saber em que condições o problema dual lagrangeano de (P) relativamente às restrições $Ax \geq b$ dado por $\max_{w \geq 0} \theta(w)$ resolve o problema (P) , isto é, em que condições $\vartheta(P) = \theta(w^*)$.

Teorema 2.1. *(Condições Suficientes de Optimalidade)*

Considere-se um par (x^*, w^*) , $x^* \in X$, $w^* \geq 0$, satisfazendo as seguintes condições

- $cx^* + w^*(b - Ax^*) = \min_{x \in X} \{cx + w^*(b - Ax)\}$
- $w^*(b - Ax^*) = 0$
- $b - Ax^* \leq 0$

então x^* é ótimo para o problema primal (P) , w^* é ótimo para o problema dual e $\theta(w^*) = \vartheta(P)$.

Demonstração. Seja x^* uma solução satisfazendo as três condições do teorema e seja x uma qualquer solução admissível de (P) , então

$$\begin{aligned} cx^* &= cx^* + w^*(b - Ax^*) \\ &\leq cx + w^*(b - Ax) \\ &\leq cx. \end{aligned}$$

A primeira igualdade ocorre porque, pela segunda condição do teorema, temos $w^*(b - Ax^*) = 0$. A primeira desigualdade é justificada usando a primeira condição do teorema e a segunda desigualdade é justificada porque $w^* \geq 0$ e porque, como x é primal admissível, temos $b - Ax \leq 0$. Deste modo, qualquer que seja a solução x primal admissível temos $cx^* \leq cx$. Além disso, usando a terceira condição do teorema podemos concluir que x^* é ótimo do primal.

Por outro lado, a primeira e segunda condições do teorema implicam que

$$\begin{aligned} \theta(w^*) &= cx^* + w^*(b - Ax^*) \\ &= cx^*. \end{aligned}$$

E como x^* é óptimo do primal $cx^* = \vartheta(P)$. Temos sempre $\theta(w^*) \leq \vartheta(P)$.

Podemos portanto concluir que w^* é óptimo do dual e $\theta(w^*) = \vartheta(P)$.

□

As condições do teorema anterior são condições suficientes de optimalidade. No entanto, no caso geral, elas não são necessárias pois pode não existir nenhum par primal-dual (x^*, w^*) que as satisfaça. O próximo teorema discute as condições suplementares que é necessário introduzir para que tal aconteça. Notamos que neste caso particular, a função objectivo cx e o conjunto de restrições relaxadas $b - Ax \leq 0$ são lineares e, portanto, não é necessário exigir condições de convexidade a estas funções.

Teorema 2.2. *(Condições Necessárias de Optimalidade)*

Se X é um convexo de \mathbb{R}^n e se a seguinte condição suplementar for verificada

- existe $x \in X$ tal que $b - Ax < 0$,

então existe x^* óptimo para o problema primal (P) e w^* óptimo para o problema dual tais que o par (x^*, w^*) satisfaz as condições do Teorema 2.1.

Caracterizada a solução óptima, interessa agora relacionar a solução óptima com o conjunto $X(w^*)$ das soluções óptimas da função dual lagrangeana para um determinado conjunto de multiplicadores óptimos. Temos o seguinte resultado.

Teorema 2.3. *Seja x^* a solução óptima de (P) , w^* a solução óptima do problema dual lagrangeano e $X(w^*)$ o conjunto das soluções óptimas da função dual lagrangeana considerando o conjunto de multiplicadores óptimos. Temos que*

$$x^* \in X(w^*)$$

Embora $x^* \in X(w^*)$ não é necessariamente um dos seus pontos extremos. Tal acontece apenas se $X(w^*)$ for um conjunto singular. Notamos que sendo x^* óptimo de (P) , é também admissível para $\theta(w^*)$. Por outro lado, usando a segunda condição do Teorema 2.1 que garante a complementaridade, isto é, $w^*(b - Ax^*) = 0$, facilmente concluímos que

$$cx^* + w^*(b - Ax^*) = cx^* = \vartheta(P)$$

e portanto x^* é também solução óptima de $\theta(w^*)$.

O valor dado pelo problema dual lagrangeano é tal que $\theta \leq \vartheta(P)$ e de um modo geral, $\theta < \vartheta(P)$. Além disso $\theta \geq \vartheta(P_L)$ pelo que se torna interessante averiguar até que ponto o limite inferior θ é melhor que o fornecido pela relaxação usual de Programação Linear.

Definição 2.2. (*Propriedade da Integralidade*)

O problema dual lagrangeano goza da propriedade da integralidade se e só se a seguinte igualdade se verifica

$$\begin{array}{ll} \theta(w) = \min \{cx + w(b - Ax)\} = \min \{cx + w(b - Ax)\} \\ \text{s.a} & \text{s.a} \\ Dx \geq f & Dx \geq f \\ x \in \bar{X} \subseteq \mathbb{Z}_+^n & x \in \text{Conv}\bar{X} \end{array}$$

onde $\bar{X} = \{0, 1\}$.

Ou seja, o dual lagrangeano goza da propriedade da integralidade se o subproblema $\theta(w)$ não se altera se tomarmos a respectiva relaxação usual de programação linear, isto é,

$$\theta(w) = \theta(w)_L.$$

Observamos que tal acontece se os subproblemas $\theta(w)$ forem, por exemplo, problemas de afectação ou fluxos em redes, de um modo geral, se forem problemas com matrizes totalmente unimodulares e cujos pontos extremos da respectiva relaxação linear são todos inteiros.

Teorema 2.4. *Se um problema dual lagrangeano goza da propriedade da integralidade, então*

$$\theta = \vartheta(P_L).$$

Demonstração. Usando a propriedade da integralidade

$$\begin{aligned}
\theta = \max_{w \geq 0} \quad & \min \{cx + w(b - Ax)\} \\
\text{s.a} \quad & Dx \geq f \\
& x \in \text{Conv} \overline{X}
\end{aligned}$$

e usando as condições de optimalidade

$$\theta = \vartheta(P_L).$$

□

Este resultado é desencorajante no que diz respeito à possibilidade de a dualidade lagrangeana produzir bons limites inferiores.

De facto, procuramos relaxações lagrangeanas tais que os subproblemas $\theta(w)$ sejam "fáceis" de resolver. Contudo, se são "fáceis" de resolver, por gozarem da propriedade da integralidade, então a dualidade lagrangeana não pode fazer melhor do que a relaxação usual de programação linear.

Por outro lado, obtêm-se "boas" relaxações lagrangeanas se evitarmos a propriedade da integralidade.

Notamos que existe uma forma ligeiramente modificada da propriedade da integralidade que é sempre verificada

$$\begin{aligned}
\theta(w) = \min \{cx + w(b - Ax)\} &= \min \{cx + w(b - Ax)\} \\
\text{s.a} & \text{s.a} \\
Dx \geq f & x \in \text{Conv} X \\
x \in \overline{X} &
\end{aligned}$$

em que a operação de convexificação é efectuada sobre todas as restrições não relaxadas. A igualdade é válida porque a função objectivo do subproblema é linear. Usando esta forma modificada temos

$$\begin{aligned}
\theta = \max_{w \geq 0} \quad & \min \{cx + w(b - Ax)\} \\
\text{s.a} & \\
& x \in \text{Conv} X
\end{aligned}$$

pelo que podemos dizer que agora a função dual lagrangeana $\theta(w)$ é a função dual de um problema de programação num conjunto convexo.

Deste modo, usando as condições necessárias de optimalidade (Teorema 2.2) podemos escrever

$$\begin{aligned} (P_{conv}) \quad & \min cx \\ & \text{s.a} \\ & Ax \geq b \\ & x \in \text{Conv}(X). \end{aligned}$$

Teorema 2.5. *O valor óptimo θ do problema dual lagrangeano de um problema (P) de programação linear inteira (ou binária) pode ser obtido resolvendo um primal modificado (P_{conv}) em que apenas se convexificam as restrições não relaxadas.*

Podemos dizer que $\theta = \vartheta(P_{conv})$ se e só se $\mathcal{F}(P_{conv}) = \text{Conv}(\mathcal{F}(P))$ (em que $\mathcal{F}(\cdot)$ representa o conjunto de soluções admissíveis do problema indicado). O que significa que dualizar, obter uma relaxação lagrangeana, é o mesmo que convexificar as restrições não relaxadas. Contudo o interesse deste teorema é apenas teórico uma vez que a tarefa de obter o poliedro convexo $\text{Conv}X$ sob a forma de um sistema de inequações em x não é fácil.

Se um problema dual lagrangeano tem a propriedade da integralidade, então

$$\theta = \vartheta(P_{conv}) = \vartheta(P_L).$$

Se por um lado é bom obter-se uma função dual lagrangeana fácil de resolver que facilite a obtenção de uma boa aproximação para o valor θ do problema dual lagrangeano. Por outro lado, muitas vezes, funções duais lagrangeanas fáceis estão associadas a problemas duais lagrangeanos que gozam da propriedade da integralidade e que portanto o seu melhor limite teórico coincide com o valor da relaxação em programação linear. Acontece, no entanto, que mesmo assim estas relaxações lagrangeanas podem e devem ser consideradas pois podem ser mais fáceis de resolver que as relaxações em programação linear. Pode, portanto, ser mais fácil obter um limite inferior

para o valor ótimo do problema através da relaxação lagrangeana do que usando a relaxação em programação linear. Este facto incentiva e justifica o grande uso das relaxações lagrangeanas mesmo nos casos em que os problemas duais lagrangeanos gozam da propriedade da integralidade.

Para aproximarmos o valor do problema dual lagrangeano e alcançarmos o melhor limite inferior possível, é necessário obtermos o valor ótimo do vector w dos multiplicadores de Lagrange. Portanto, podemos usar, entre outros, métodos de optimização por subgradiente, por subgradiente radar e por feixe de subgradientes, que se descrevem no capítulo seguinte.

Capítulo 3

Métodos de resolução da relaxação lagrangeana

Neste capítulo estudamos métodos de resolução do problema dual lagrangeano

$$\max_{w \geq 0} \theta(w)$$

que têm como objectivo determinar uma sequência de soluções duais que aproximam o valor da sua solução óptima. A solução do problema dual fornece um limite inferior para o valor óptimo do problema que pode ser bastante útil em esquemas de optimização.

Recordamos que a função dual lagrangeana $\theta(w)$ é uma função côncava e linear por partes, portanto não é diferenciável. Contudo, é subdiferenciável e um dos seus subgradientes em w pode ser facilmente obtido depois de se resolver a correspondente relaxação lagrangeana $\min_{x \in X} \{cx + w(b - Ax)\}$.

De um modo geral, os métodos usados determinam iterativamente para $t = 0, 1, \dots$ um conjunto de multiplicadores w_t usando uma determinada direcção d_t , que se pretende seja ascendente, um tamanho de passo p_t e actualizam $w_{t+1} := w_t + p_t d_t$. Neste tipo de métodos há que ter cuidado com o cálculo do subgradiente e da função, pois muitas vezes esse cálculo é aproximado. Além disso, há que ter em atenção que uma pequena variação no valor de w_t pode provocar grandes variações no subgradiente. Existem dois métodos principais que maximizam a função dual lagrangeana, o método do subgradiente e o método de planos de corte. Poderemos mesmo dizer que constituem dois grupos/famílias de métodos uma vez que cada um deles possui as suas muitas variantes.

O método de planos de corte tem usualmente um comportamento lento e instável junto do óptimo pelo que é usualmente usado na forma de feixe (método do feixe de subgradientes) onde uma penalidade quadrática é considerada na função objectivo. Neste grupo de métodos podemos também considerar o método subgradiente radar que descreveremos.

Apesar das desvantagens do método subgradiente é ainda um dos métodos mais populares pois é simples de implementar e computacionalmente pouco complexo. Além disso, apesar de a sua taxa de convergência ser fraca (linear para o método subgradiente puro) a experiência tem mostrado que a sua eficiência prática é bastante boa, especialmente para problemas de optimização de grande dimensão e com algum tipo de estrutura. Por este motivo o método do subgradiente é ainda objecto de investigação activa e por isso é também o método sobre o qual o nosso trabalho mais incide.

Embora distintos todos os métodos têm a mesma estrutura geral que pode ser descrita da seguinte forma:

Algoritmo para optimização da função dual lagrangeana

Passo 0: (inicialização)

considerar $\epsilon > 0$,

escolher $w_0 \geq 0$ e fazer $t = 0$

Passo 1: (resolução da relaxação)

obter a solução óptima x_t de $\theta(w_t)$, seja θ_t o seu valor e calcular o subgradiente

$$g_t = g(x_t)$$

Passo 2: (cálculo da direcção e do tamanho do passo)

obter a direcção d_t e o tamanho do passo p_t

Passo 3: (actualização dos multiplicadores)

fazer $w_{t+1} = w_t + p_t d_t$

Passo 4: (critério de paragem)

se w_t não melhora nas últimas iterações ou t alcança um valor pré-fixado

ou $\|w_{t+1} - w_t\| < \epsilon$, STOP

senão fazer $t = t + 1$ e voltar ao Passo 1.

Por uma questão de facilidade na descrição dos métodos, consideramos que $g(x) = b - Ax$, sendo portanto $g(x)$ o vector subgradiente de θ em w e definimos $g_t = g(x_t)$.

No Passo 0 temos de inicializar todos os parâmetros que serão usados no algoritmo e temos de fornecer um valor inicial para o conjunto dos multiplicadores w .

No Passo 1 efectua-se a optimização da função dual lagrangeana para os multiplicadores correntes, w_t , com obtenção do seu valor e da sua solução x_t e calcula-se o subgradiente, g_t . Este passo do algoritmo será específico de cada problema e não do método pois trata-se fundamentalmente da optimização da função dual lagrangeana que é específica de cada problema.

Descreveremos de seguida o Passo 2 dos métodos subgradiente, subgradiente radar e feixe de subgradientes nos quais para um determinado conjunto de multiplicadores w calculamos o valor da função $\theta(w)$ e também um subgradiente arbitrário de θ em w . Estes métodos diferem essencialmente na forma como usam a informação disponível para o cálculo da direcção e do tamanho do passo. Para o fazer na resolução do problema dual lagrangeano têm ao seu dispor apenas o vector w dos multiplicadores de Lagrange, o valor $\theta(w)$ da função dual lagrangeana e o vector $g(x)$ do subgradiente. É, portanto, apenas com esta informação que se pretende obter um esquema de aproximação do valor óptimo do problema dual lagrangeano usando uma direcção ascendente que é difícil de obter dada a pouca informação existente. Vejamos como cada um dos métodos descritos lida com a informação disponível para atingir o objectivo de aproximar o valor da função dual.

3.1 Método subgradiente

O subgradiente de uma função não diferenciável e côncava não é necessariamente uma direcção ascendente, e o seu subdiferencial não é conhecido na sua totalidade. Mesmo assim o objectivo do método subgradiente [20, 21] é o de tentar copiar o comportamento ascendente definido pela direcção do gradiente.

Dado um valor inicial w_0 ao vector dos multiplicadores w , na iteração $t+1$, podemos determinar um novo ponto $w_{t+1} = w_t + p_t d_t$ sendo p_t o tamanho do passo e a direcção

$d_t = g(w_t)$, obtida de modo semelhante à direcção do método gradiente. A direcção a calcular no Passo 2 deve ser ascendente, se a função $\theta(w)$ é côncava. Contudo nos casos em que a função não é diferenciável não é possível garantir tal condição. Várias são as regras propostas para o cálculo da direcção.

Regra de *Held, Wolfe & Crowder* (cf. [16])

$$d_t = g_t$$

Regra de *Crowder* (cf. [7])

$$d_t = g_t + \mu d_{t-1}$$

em que μ deve ser fixo a um valor < 1 sendo $\mu = 0.5$ um valor muito comumente usado.

Regra de *Camerini, Fratta, Maffioli* (cf. [5])

$$d_t = \begin{cases} g_t - \mu \frac{g_t d_{t-1}}{\|d_{t-1}\|^2} d_{t-1}, & \text{se } g_t d_{t-1} < 0 \\ g_t, & \text{caso contrário} \end{cases}$$

em que $0 \leq \mu \leq 2$ deve ser determinado através de experiências computacionais sendo $\mu = 1.5$ considerado uma boa escolha indicada em [5]. Considerando-se $\mu = -\frac{\|g_t\| \|d_{t-1}\|}{g_t d_{t-1}}$ obtém-se a seguinte regra.

Regra de *Camerini, Fratta, Maffioli modificada* (cf. [5])

$$d_t = \begin{cases} g_t + \frac{\|g_t\|}{\|d_{t-1}\|} d_{t-1}, & \text{se } g_t d_{t-1} < 0 \\ g_t, & \text{caso contrário} \end{cases}$$

Notamos que as primeiras versões do método subgradiente [16, 20] usavam apenas o subgradiente no cálculo da direcção: regra *Held, Wolf & Crowder* e regra de *Crowder*. Contudo, em versões mais recentes [5, 7], na regra de *Camerini, Fratta, Maffioli* é também tomada em consideração a direcção usada na iteração anterior, uma vez que pode levar a melhoramentos no desempenho do algoritmo.

Uma escolha adequada do tamanho do passo p_t produz um método convergente. Este tamanho de passo deve ser tal que $\theta(w_t + p_t d_t) < \theta(w_t)$ e é habitual usar-se

$$p_t = \rho \frac{(\theta^* - \theta(w_t))}{g_t d_t}$$

com $0 < \rho \leq 2$, sendo θ^* um valor dado que, idealmente, deveria ser o valor óptimo, contudo, na sua ausência usa-se um limite superior para o valor da função. Estes dois parâmetros são críticos para o bom funcionamento do método subgradiente.

Notamos que alguns resultados teóricos relacionados com uma escolha apropriada do tamanho do passo [16] consideram que $\theta(w_t)$ convergirá para o valor óptimo se $\lim_{t \rightarrow +\infty} p_t = 0$ e $\sum_{t=0}^{\infty} p_t = +\infty$ e a sugestão é a de usar a expressão anterior.

Sendo a função dual não diferenciável, o método subgradiente pode progredir lentamente para o óptimo de um modo oscilante.

O método do subgradiente é um método simples e tem muitas desvantagens, contudo, devido à sua simplicidade, é um dos métodos mais usados na resolução do problema dual lagrangeano.

Na Tabela 3.1 mostramos a forma de obter o próximo ponto consoante a regra usada para a determinação da direção no método subgradiente.

Tabela 3.1: Cálculo do próximo ponto no método subgradiente

Regras	
<i>Held, Wolfe & Crowder</i>	$w_{t+1} = w_t + \rho \frac{(\theta^* - \theta(w_t))}{g_t d_t} g_t$
<i>Crowder</i>	$w_{t+1} = w_t + \rho \frac{(\theta^* - \theta(w_t))}{g_t d_t} (g_t + 0.5 d_{t-1})$ $= w_t + \rho \frac{(\theta^* - \theta(w_t))}{g_t}$
<i>Camerini, Fratta, Maffioli</i>	$w_{t+1} = w_t + \rho \frac{(\theta^* - \theta(w_t))}{g_t d_t} d_t \text{ em que}$ $d_t = \begin{cases} g_t - \mu \frac{g_t d_{t-1}}{\ d_{t-1}\ ^2} d_{t-1}, & \text{se } g_t d_{t-1} < 0 \\ g_t, & \text{caso contrário} \end{cases}$
<i>Camerini, Fratta, Maffioli modificada</i>	$w_{t+1} = w_t + \rho \frac{(\theta^* - \theta(w_t))}{g_t d_t} d_t$ $d_t = \begin{cases} g_t + \frac{\ g_t\ }{\ d_{t-1}\ } d_{t-1}, & \text{se } g_t d_{t-1} < 0 \\ g_t, & \text{caso contrário} \end{cases}$

3.2 Método subgradiente radar

O método subgradiente radar [1, 2, 3] é uma adaptação do método radar para a optimização de funções não diferenciáveis tal como o método do subgradiente o é relativamente ao método do gradiente. O método subgradiente radar assume que não se conhece uma expressão explícita para a função dual mas é conhecida uma aproximação de primeira ordem e assume que é sempre possível melhorar a solução actual seguindo a direcção do subgradiente. No caso em que a função dual não é diferenciável podemos obter uma aproximação de primeira ordem usando o conjunto de subgradientes calculados para construir planos de suporte. Contudo, o subgradiente pode não ser uma direcção ascendente, pelo que algumas estratégias têm de ser usadas para contornar este problema. O objectivo do método subgradiente radar é, portanto, o de maximizar qualquer função não diferenciável e côncava $\theta(w)$ sem restrições executando uma procura ao longo da direcção dada pelo subgradiente e usando uma aproximação de primeira ordem da função $\theta(w)$ de modo a calcular o tamanho do passo ao longo da direcção. O tamanho do passo é calculado ao longo da direcção do seguinte modo:

- se não existe nenhum plano de suporte tangente de paragem, então move-se o ponto w_t ao longo da direcção subgradiente, dá-se um passo;
- se existe pelo menos um plano tangente de paragem, então move-se o ponto w_t apenas até ao primeiro plano de suporte tangente de paragem ao longo da direcção do subgradiente.

Dado um valor inicial w_0 ao vector dos multiplicadores w , na iteração $t + 1$ podemos determinar um novo ponto $w_{t+1} = w_t + p_t d_t$ sendo p_t o tamanho do passo e d_t a direcção. Neste processo iterativo de optimização de $\theta(w)$, obtemos os valores das funções $\theta_s = \theta(w_s)$ e os subgradientes g_s , $s = 1, \dots, t$, que permitem construir planos de suporte PS_1, \dots, PS_t , tais que: $PS_s = \theta_s + g_s(w - w_s)$, $s = 1, \dots, t$. O envelope convexo inferior dos planos de suporte fornece-nos uma aproximação de primeira ordem para $\theta(w)$, isto é, $\tilde{\theta}_t(w) = \tilde{\theta}_t = \min_{s=1, \dots, t} \{\theta_s + g_s(w - w_s)\}$.

A ideia geométrica, dada de uma forma intuitiva, e a forma como é determinado o tamanho do passo no método subgradiente radar está exemplificada na Figura 3.1.

Consideramos o caso unidimensional e suponhamos que a aproximação de primeira ordem para a função $\theta(w)$ é baseada em três planos de suporte PS_t, PS_{t-1} e PS_{t-2} . Neste caso, para calcular o tamanho do passo de w_t para w_{t+1} , calculam-se primeiro os pontos $y_{t-1} := PS_t \cap PS_{t-1}$ e $y_{t-2} := PS_t \cap PS_{t-2}$. Em segundo lugar, projectamos y_{t-1} e y_{t-2} na direcção subgradiente e obtemos $q_{t-1} := \text{proj}(y_{t-1})$ e $q_{t-2} := \text{proj}(y_{t-2})$. Em terceiro lugar, w_{t+1} é definido como a projecção mais próxima de w_t , isto é, q_{t-2} .

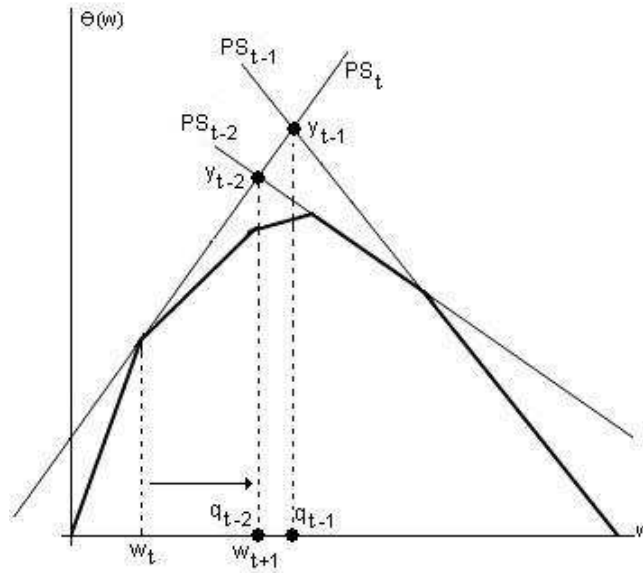


Figura 3.1: Tamanho do passo

Contudo há que acautelar algumas situações. Visto que a direcção subgradiente pode conduzir a uma iteração "má" porque a direcção pode não ser ascendente, tal como anteriormente, temos de calcular a intersecção de todos os planos de suporte anteriores com PS_t , projectar essas intersecções na direcção subgradiente e tomar w_{t+1} como a projecção mais próxima de w_t . A diferença em relação ao caso anterior, é que agora não se tem em conta PS_{t-1} porque o seu declive tem o mesmo sinal que o declive de PS_t . Se consideramos planos de suporte com o sinal do declive de PS_t o processo pode parar demasiado cedo antes de alcançar o óptimo. Deste modo, definimos $m_s := \frac{g_s g_t}{\|g_t\|}$, $s = 1, \dots, t-1$ como sendo o declive do plano de suporte PS_s ao longo da direcção subgradiente. Dizemos que o plano de suporte PS_s definido pelo ponto (w_s, θ_s) e o subgradiente g_s , $s = 1, \dots, t-1$, é um plano positivo relativo a w_t se a linha de

procura tem declive positivo, isto é, se $m_s > 0$. Se m_s não é maior que zero, dizemos que PS_s é um plano negativo relativo a w_t . Em termos computacionais é suficiente calcular $g_s g_t$ para determinar se o plano é positivo ou negativo. De seguida, ver Figura 3.2, calculamos a intersecção de todos os planos de suporte com o plano PS_t , projectamos estas intersecções na direcção subgradiente e tomamos w_{t+1} como sendo a projecção mais próxima de w_t de entre os planos com declive contrário ao de PS_t . Devemos, portanto, evitar planos com declive positivo.

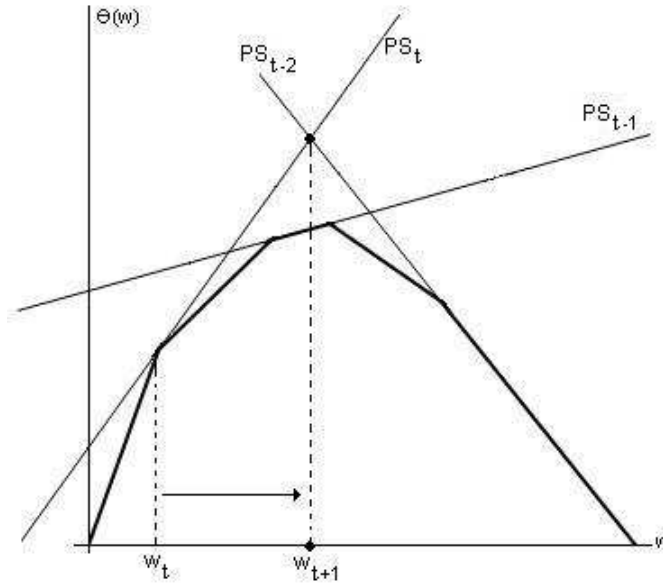


Figura 3.2: Planos positivos e negativos

A intenção deste método é a de detectar e evitar planos positivos. Contudo, se a aproximação de primeira ordem de $\theta(w)$ ao longo da direcção subgradiente é baseada apenas num conjunto de planos positivos (isto é, todos os planos têm o sinal do declive de PS_t e não há planos negativos relativamente a w_t), então o tamanho do passo radar de w_t para w_{t+1} não pode ser calculado e portanto usamos arbitrariamente o tamanho do passo usado no método subgradiente clássico.

No Passo 2 do método subgradiente radar a direcção é obtida fazendo

$$d_t = g_t$$

e o tamanho do passo p_t é calculado do seguinte modo

- para $s = 0, \dots, t - 1$

- calcular $g_s g_t$

- se $g_s g_t > 0$ rejeitar PS_s

senão determinar o ponto de intersecção definido por

$$\beta_{ts} := \frac{\theta_s - \theta_t + (w_t - w_s)g_s}{g_t g_t - g_s g_t}$$

- determinar o conjunto dos planos positivos $\Omega = \{\beta_{ts} > 0, s = 0, \dots, t - 1\}$

- se $\Omega \neq \emptyset$ então o tamanho do passo $p_t = \min \Omega$

$$\text{senão } p_t = \frac{\rho}{\|g_t\|} \text{ ou } p_t = \rho \frac{(\theta^* - \theta(w_t))}{g_t d_t}$$

Aquando da determinação do tamanho do passo usamos $p_t = \min \Omega$ ou então $p_t = \frac{\rho}{\|g_t\|}$ ou $p_t = \rho \frac{(\theta^* - \theta(w_t))}{g_t d_t}$ que é o passo usado no método subgradiente. Observamos que o método subgradiente radar pode ser melhorado quando se determina uma solução óptima para o problema da procura do tamanho do passo [3], podendo-se efectuar uma procura exacta substituindo a forma como se determina o tamanho do passo quando $\Omega \neq \emptyset$ do seguinte modo:

$$p_t = \min \left\{ \beta_{ts} \text{ tal que } \beta_{ts} = \frac{\theta_s - \theta_t + (w_t - w_s)g_s}{g_t g_t - g_s g_t}, \beta_{ts} > p_{t-1}, s = 0, \dots, t - 1 \right\}.$$

E portanto, em vez de considerar todos os tamanhos de passo positivos, usa-se o melhor tamanho de passo determinado na iteração anterior.

3.3 Método do feixe de subgradientes

Nesta secção descreveremos o método do feixe de subgradientes [4, 6, 11] que tem sido já bastante estudado mas cuja aplicação a problemas de programação inteira ainda é pouco utilizada. É um método com melhores propriedades que os anteriores contudo mais complicado de usar.

A ideia principal do método do feixe de subgradientes é a de, como no método subgradiente radar, usar informação obtida ao longo das iterações para construir um

modelo da função dual lagrangeana $\theta(w)$ que se pretende maximizar (contudo fá-lo de forma diferente que no método subgradiente radar) e usar este modelo para guiar a procura do valor óptimo usando uma direcção com propriedades ascendentes e de estabilidade, o que não acontece com o método subgradiente radar.

O método do subgradiente não garante o aumento do valor da função dual em cada iteração uma vez que a direcção fornecida pelo subgradiente da função dual não é necessariamente uma direcção ascendente. Para construir uma direcção ascendente seria necessário mais informação que o simples conhecimento de um subgradiente. O método subgradiente radar não garante a estabilidade da direcção, podendo o novo ponto calculado estar "demasiado longe" do anterior. O método do feixe de subgradiantes que agora descrevemos guarda um conjunto de subgradiantes, um feixe, da função dual lagrangeana obtidos em iterações anteriores e, para além disso, combina simultaneamente propriedades ascendentes e de estabilidade da direcção. Contudo este melhoramento traduz-se num aumento da complexidade do método.

Tal como os dois métodos já descritos, também o método do feixe de subgradiantes consiste em dois passos essenciais: primeiro encontrar uma direcção ascendente e depois proceder à escolha do tamanho do passo a considerar. No primeiro passo há que determinar um modelo para a função $\theta(w)$ que tenha em conta toda a informação obtida ao longo das iterações anteriores e que permita encontrar uma direcção ascendente. No segundo passo, como consequência da escolha do tamanho do passo, terá de ser determinado o novo conjunto de multiplicadores. Esta acção resulta num *passo autêntico* (quando o novo ponto ao longo da direcção produz um aumento suficiente no valor da função dual lagrangeana) ou num *passo nulo* (caso contrário).

Para impedir a função dual lagrangeana $\theta(w)$ de decrescer de valor seria desejável que fosse retida informação acerca do melhor conjunto de multiplicadores obtido até ao momento. Guardando esta informação ao longo das iterações $t = 0, 1, \dots$, o algoritmo deve ser capaz de gerar duas sequências de pontos. Uma é a sequência de pontos usada para definir o modelo, $\theta_{\mathcal{B}}(y)$, da função dual lagrangeana. Chamamos a estes pontos de *multiplicadores candidatos* e são denotados por y^t . A segunda sequência é formada pelos multiplicadores candidatos que aumentam suficientemente o valor da função dual

lagrangeana. A estes pontos chamamos de *centros de estabilidade* e denotamo-los por u^t . Note que $\{u^t\}$ é uma subsequência de $\{y^t\}$.

Há muitas formas de definir um modelo para a função $\theta(w)$ que gera multiplicadores (centros de estabilidade) u^t para os quais $\theta(u^t)$ é o melhor valor obtido até ao momento. Essencialmente o modelo definido deve prevenir movimentos para "muito longe" de w_t . Tal como no método subgradiente radar, se a informação obtida ao longo das iterações for sendo guardada, é possível construir um modelo de planos de corte da função $\theta(w)$ que, para $k \in \mathcal{B} = \{1, \dots, t\}$ usa os valores de $\theta_k = \theta(w_k)$ e dos subgradientes g_k obtidos até ao momento (iteração t). A seguinte função, afim por partes,

$$\theta_{\mathcal{B}} = \theta_{\mathcal{B}}(y) = \min_{k \in \mathcal{B}} \{\theta(w_k) + g_k(y - w_k)\}$$

é uma aproximação da função $\theta(w)$.

Depois é usado o vector y maximizante de $\theta_{\mathcal{B}}(y)$ como sendo o próximo conjunto de multiplicadores w_{t+1} . Portanto, o algoritmo do método deve iterativamente determinar y , o maximizante do modelo $\theta_{\mathcal{B}}(y)$, verificar se esse valor provoca um aumento no valor da função dual, isto é, se

$$\theta(y) - \theta(w_t) \geq tol,$$

onde $tol \geq 0$ é um parâmetro dado, e se tal acontecer efectuar um *passo autêntico* $w_{t+1} = y$, se não, efectuar um *passo nulo* $w_{t+1} = w_t$.

Notamos que, por definição, o modelo $\theta_{\mathcal{B}}(y)$ construído é uma aproximação superior da função $\theta(w)$. Se $\mathcal{B}' = \mathcal{B} \cup \{t+1\}$, então

$$\theta_{\mathcal{B}'} \leq \theta_{\mathcal{B}} \quad \text{e} \quad \theta(w) \leq \theta_{\mathcal{B}}(y)$$

(uma vez que sendo g_k um subgradiente de θ em w_k se tem $\theta(w) \leq \theta(w_k) + g_k(w - w_k)$).

Notamos também que $\theta_{\mathcal{B}}$ pode ser reescrita em função da direcção $d = y - w_t$ obtendo $\theta_{\mathcal{B},d} = \theta_{\mathcal{B},d}(d) = \theta_{\mathcal{B}}(w_t + d) = \min_{k \in \mathcal{B}} \{\beta_k + g_k d\} + \theta(w_t)$ sendo $\beta_k = \theta(w_k) + g_k(w_t - w_k) - \theta(w_t) \geq 0$ o erro de linearização.

Contudo, este trata-se do método simples de planos de corte e tem algumas desvantagens. Quando ainda são conhecidos poucos subgradientes o maximizante de $\theta_{\mathcal{B}}$ está muito provavelmente pouco definido. Outra desvantagem é a de que pode gerar pontos

muito afastados uns dos outros (mesmo que um ponto esteja próximo do óptimo, o ponto da iteração seguinte pode estar longe), ou seja não tem propriedades de estabilidade locais. No entanto essas condições podem ser impostas. É a imposição dessas condições que caracteriza o método de feixe de subgradientes e o distingue do clássico método de planos de corte. Desta forma, o método do feixe de subgradientes pode ser visto como uma versão estabilizada do método de planos de corte em que a função $\theta_{\mathcal{B}}$ é maximizada sujeita a condições de estabilidade local.

Nesse sentido há que considerar uma forma de seleccionar o novo ponto w_{t+1} que não seja "demasiado longe" do anterior. Há muitas formas de efectuar a selecção da direcção, e do tamanho do passo, satisfazendo esta condição. Uma delas descreve-se de seguida em que a estabilização resulta da introdução de um termo quadrático no modelo de planos de corte usado para a função $\theta(w)$. O próximo ponto pode ser seleccionado através da resolução do seguinte problema quadrático

$$\bar{\theta}_{\mathcal{B}}(y) = \max_y \{ \theta_{\mathcal{B}}(y) - \frac{1}{r_t} \|y - w_t\| \}$$

ou, equivalentemente, tendo em conta que $d = y - w_t$, a direcção pode ser seleccionada através da resolução do seguinte problema quadrático

$$(QP) \quad \bar{\theta}_{\mathcal{B}} = \bar{\theta}_{\mathcal{B},d}(d) = \max_d \{ \theta_{\mathcal{B},d}(d) - \frac{1}{2r_t} \|d\|^2 \}$$

em que $r_t > 0$ é chamado de parâmetro da região de confiança. O problema anterior atinge sempre uma única solução finita d que será usada como a próxima direcção com tamanho de passo constante e igual a 1. Observamos que r_t pode ser também interpretado como o tamanho de passo. O dual quadrático de (QP) é:

$$\min_{\lambda \geq 0} \left\{ \frac{r_t}{2} \left\| \sum_{j \in \mathcal{B}} g_j \lambda_j \right\|^2 + \sum_{j \in \mathcal{B}} \alpha_j \lambda_j \mid \sum_{j \in \mathcal{B}} \lambda_j = 1 \right\}$$

e $d_t = r_t z_t$ onde $z_t = \sum_{j \in \mathcal{B}} g_j \lambda_j^t$ e λ^t é a solução óptima do dual do problema quadrático. Desta forma o novo ponto será encontrado ao longo da direcção z_t que é uma combinação convexa dos subgradientes anteriores e usando r_t como tamanho de passo.

O Passo 2 do algoritmo do método do feixe de subgradientes pode ser descrito do seguinte modo.

Passo 2: (cálculo da direcção e do tamanho do passo)

- fazer $p_t = 1$
- obter a direcção d_t do seguinte modo
 - obter a solução óptima d do seguinte problema quadrático

$$\max_d \{ \theta_{\mathcal{B},d}(d) - \frac{1}{2r_t} \|d\|^2 \}$$
 - se $\theta(w_t + d) - \theta(w_t) \geq tol$ então $d_t = d$ senão $d_t = 0$

Portanto, o método do feixe de subgradientes requer em cada iteração a solução, no Passo 2, de um problema não linear para calcular o próximo conjunto de multiplicadores. Esta torna-se uma grande desvantagem deste método pois é necessário usar códigos especializados de programação quadrática de modo a obter-se uma eficiente implementação do método. Os problemas quadráticos (QP) são essencialmente resolvidos através do seu dual que possui uma estrutura especial. Bons "QP solvers" exploram esta estrutura e permitem ainda comprimir a informação do feixe usando técnicas de agregação. Algoritmos especializados na resolução do problema quadrático são fundamentais e, em particular quando há a necessidade adicional, como é o caso, de considerar restrições adicionais. Além disso há ainda a necessidade de garantir a admissibilidade do conjunto de multiplicadores da próxima iteração, isso significa que teremos de adicionar ao problema (QP) o seguinte conjunto de restrições $y \geq 0$ ou equivalentemente $d \geq -w_t$ tornando-o mais difícil de resolver.

Notamos que a informação do feixe consiste nos seguintes elementos $(w_k, \theta(w_k), g(x_k))$ e quando o feixe se torna muito grande este deve ser comprimido. Portanto, uma possibilidade para reduzir o esforço da resolução do problema (QP) é a de manter pequeno o tamanho do feixe de subgradientes. Usualmente os subgradientes podem ser abandonados quando estão inactivos durante um determinado número de iterações consecutivas. É também possível estabelecer um tamanho máximo (≥ 2) para o feixe, com a condição de que o subgradiente agregado z_t (e o seu correspondente erro de linearização, $\sigma_t = \sum_{j \in \mathcal{B}} \alpha_j \lambda_j^i$) seja adicionado ao feixe cada vez que um subgradiente "activo" é eliminado. Há que ponderar esta limitação pois feixes pequenos podem originar

problemas de convergência embora reduza significativamente o esforço computacional do algoritmo.

A juntar a todas estas particularidades deste método, há também que considerar a escolha do parâmetro r_t da região de confiança, que pode ser vista como a escolha do tamanho do passo. Esta escolha é muito importante no desempenho do algoritmo e está ligada à obtenção do novo ponto w_{t+1} , uma vez que pode ser dado um *passo autêntico* ou um *passo nulo*. O ponto w_{t+1} é distinto do anterior (é dado um *passo autêntico*) quando o valor $\theta(w_{t+1})$ é melhor que o de $\theta(w_t)$, ou seja, $\theta(w_{t+1}) - \theta(w_t) \geq m_1 \bar{\theta}_{\mathcal{B}}(d_t)$ para um determinado parâmetro fixo $0 \leq m_1 < 1$. Notamos que $\bar{\theta}_{\mathcal{B}}(d_t)$ é o melhoramento previsto pelo modelo de planos de corte para um movimento de w_t para w_{t+1} . Portanto, quando o modelo de planos de corte na iteração actual é bom para a função $\theta(w)$ significa que faremos um *passo autêntico*. Por outro lado, quando o modelo que se tem para $\theta(w)$ é fraco efectua-se um *passo nulo* e nova informação é recolhida para melhorar $\bar{\theta}_{\mathcal{B}}$ como modelo de $\theta(w)$ de modo a obtermos uma melhor direcção d_{t+1} . Podemos escolher r_t com um valor fixo ao longo de todas as iterações ou então outras regras podem ser escolhidas (ver [11] onde outras regras são descritas).

Embora haja neste algoritmo a complexidade acrescida pela necessidade de resolução de um problema quadrático, este é o preço a pagar pela obtenção de estabilidade, e usualmente os problemas quadráticos não são muito mais difíceis de resolver que os problemas lineares existindo vários códigos capazes de os resolver eficientemente.

Tal como o método subgradiente radar, o método do feixe de subgradientes usa um modelo de planos de corte para a função dual contudo, exige propriedades de estabilidade local, através da inclusão de um termo quadrático no modelo.

Capítulo 4

O problema HMST e uma sua relaxação lagrangeana

Neste capítulo começamos por definir o problema HMST e apresentamos uma formulação proposta por Gouveia [13, 14], a formulação de fluxos orientada com índices de salto (HMCF). Depois construímos uma relaxação lagrangeana e detalhamos o modo de resolução do problema dual lagrangeano tendo em conta o uso dos métodos descritos no Capítulo 3.

4.1 O problema HMST

O Problema da Árvore de Suporte de Custo Mínimo com Restrições de Salto, HMST (*Hop-constrained Minimum Spanning Tree*), é definido da seguinte forma:

Seja $\mathcal{G} = (\mathcal{N}, \mathcal{E})$ um grafo simples (isto é, um grafo sem arestas paralelas e lacetes), onde $\mathcal{N} = \{0, 1, \dots, n\}$ representa o conjunto dos nodos e $\mathcal{E} = \{\{i, j\} : i, j \in \mathcal{N}, i \neq j\}$ o conjunto das arestas, tal que a cada aresta $\{i, j\} \in \mathcal{E}$ está associado um custo c_{ij} . Seja ainda, H um número natural.

Dado um nodo raíz (que sem perda de generalidade assumimos ser o nodo 0), o Problema da Árvore de Suporte de Custo Mínimo com Restrições de Salto consiste em determinar no grafo \mathcal{G} a árvore de suporte, T , de custo mínimo (sendo o custo de T dado por $\sum_{\{i,j\} \in T} c_{ij}$) com a restrição adicional de o número de saltos

(arestas) no único caminho desde o nodo raiz até qualquer outro nodo da árvore não ser superior a H .

Considere-se um grafo para o qual se pretende obter a árvore de suporte de custo mínimo com a restrição adicional de o número de arestas no único caminho entre o nodo raiz (nodo 0) e os restantes nodos do grafo não ser superior a 3, ou seja, com $H = 3$. Admita-se que se aplicou o algoritmo de Prim e se obteve a árvore de suporte de custo mínimo que se apresenta na Figura 4.1.

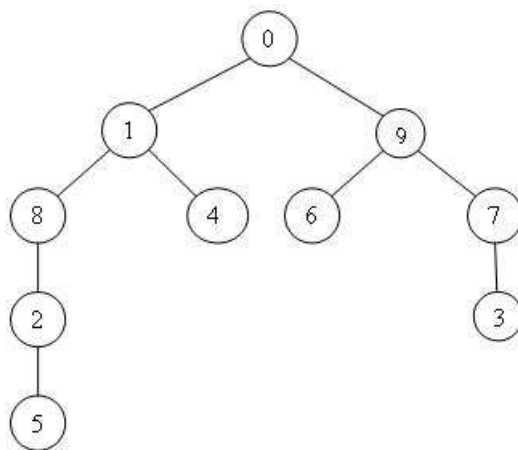


Figura 4.1: Árvore de Suporte de Custo Mínimo

Note que $[0, 1, 8, 2, 5]$ é um caminho com 4 arestas pelo que esta solução não é admissível para o nosso problema pois viola a restrição de salto entre o nodo 0 e o nodo 5. Uma solução admissível seria, por exemplo, a árvore da Figura 4.2.

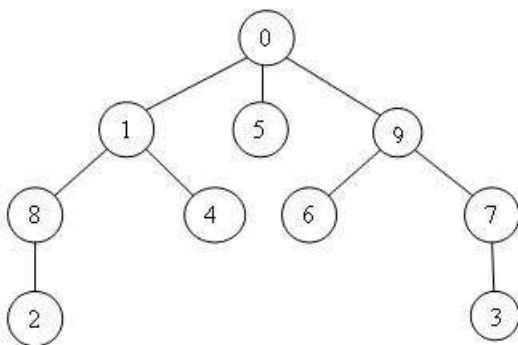


Figura 4.2: Árvore de Suporte de Custo Mínimo com restrições de salto para $H = 3$

4.2 Formulação de fluxos orientada com índices de salto

Nesta secção vamos apresentar uma formulação para o problema HMST. Esta formulação é uma formulação de fluxos e foi obtida por Gouveia [14] seguindo a técnica da redefinição de variáveis, será designada por HMCF, e utiliza variáveis de fluxo com índices de salto.

Recentes avanços em Optimização Combinatória indicam que uma melhor formulação (mais compacta ¹ e/ou com melhor valor da relaxação linear), para vários problemas de redes, pode ser obtida, definindo o problema num grafo orientado (veja-se, por exemplo, Magnanti e Wolsey [17]). Além disso, formulações orientadas, isto é, definidas em grafos orientados, têm-se mostrado mais adequadas para utilizar em processos de obtenção de limites inferiores [13]. Por esta razão, neste trabalho, apenas consideramos uma formulação orientada para o problema HMST e que foi considerada uma boa formulação para o problema [14, 15].

Um problema definido num grafo não orientado $\mathcal{G} = (\mathcal{N}, \mathcal{E})$ pode facilmente ser transformado num problema equivalente num grafo orientado, substituindo cada aresta $e = \{i, j\} \in \mathcal{E}$, por dois arcos, o arco (i, j) e o arco (j, i) , e associando a cada um destes novos arcos o custo da aresta que lhes deu origem. Assim, o grafo resultante é orientado e simétrico, isto é, $c_{ij} = c_{ji}$, $\forall i, j \in \mathcal{N}, i \neq j$. O problema de encontrar a árvore de suporte de custo mínimo é então equivalente ao problema de encontrar a arborescência de custo mínimo, com raiz num qualquer nodo, no grafo orientado equivalente [17, 19]. Para simplificar e sem perda de generalidade, assumimos que o nodo raiz é o nodo 0 [17, 19]. Também assumimos que os arcos da arborescência são dirigidos para fora do nodo raiz, consequentemente, cada aresta $\{0, j\} \in \mathcal{E}$ é apenas substituída por um único arco, o arco $(0, j)$.

No que se segue designaremos por $\mathcal{D} = \mathcal{N} \setminus \{0\}$ o conjunto de todos os nodos de \mathcal{G} excepto o nodo raiz, por \mathcal{A} o conjunto dos arcos no grafo orientado, isto é, $\mathcal{A} = \{(i, j) :$

¹Dizemos que uma formulação é compacta quando envolve um número polinomial de variáveis e de restrições.

$i \in \mathcal{N}, j \in \mathcal{D}, i \neq j\}$.

A formulação de fluxos com índices de salto para o HMST utiliza as variáveis binárias x_{ij} , $(i, j) \in \mathcal{A}$, que indicam se o arco (i, j) pertence ou não à árvore de suporte (ou melhor, arborescência) e as variáveis de fluxo z_{ij}^{hk} , $(i, j) \in \mathcal{A}, k \in \mathcal{D}, i \neq k, h = 1, \dots, H$, que indicam se o arco (i, j) é ou não incluído exactamente na posição h no único caminho da raiz para o nodo k , ou seja:

$$x_{ij} = \begin{cases} 1, & \text{se o arco } (i, j) \text{ pertence à arborescência de custo mínimo} \\ 0, & \text{caso contrário} \end{cases}$$

$$z_{ij}^{hk} = \begin{cases} 1, & \text{se o arco } (i, j) \text{ é incluído exactamente na posição } h \text{ no único} \\ & \text{caminho da raiz para o nodo } k \\ 0, & \text{caso contrário} \end{cases}$$

Notamos que as variáveis x_{ij} e z_{ij}^{hk} podem ser consideradas como sendo variáveis orientadas uma vez que estão associadas a arcos, e não a arestas, pelo que $x_{ij} \neq x_{ji}$ e $z_{ij}^{hk} \neq z_{ji}^{hk}$, $\forall i, j \in \mathcal{N}$.

A nova formulação para o problema HMST tem a seguinte forma.

Formulação (HMCF):

$$\min \sum_{(i,j) \in \mathcal{A}} c_{ij} x_{ij}$$

sujeito a

$$\sum_{i \in \mathcal{N}} x_{ij} = 1, \quad j \in \mathcal{D} \quad (1)$$

$$z_{0i}^{1k} - \sum_{j \in \mathcal{N}} z_{ij}^{2k} = 0, \quad i, k \in \mathcal{D} \quad (2)$$

$$\sum_{j \in \mathcal{N}} z_{ji}^{hk} - \sum_{j \in \mathcal{N}} z_{ij}^{h+1,k} = 0, \quad i, k \in \mathcal{D}, h = 2, \dots, H-1 \quad (3)$$

$$\sum_{j \in \mathcal{N}} z_{jk}^{Hk} = 1, \quad k \in \mathcal{D} \quad (4)$$

$$\sum_{h=1}^H z_{ij}^{hk} \leq x_{ij}, \quad (i, j) \in \mathcal{A}, k \in \mathcal{D} \quad (5)$$

$$z_{0j}^{1k} \in \{0, 1\} \quad j, k \in \mathcal{D} \quad (6)$$

$$z_{ij}^{hk} \in \{0, 1\} \quad (i, j) \in \mathcal{A}, k \in \mathcal{D}, i \neq 0, k, h = 2, \dots, H \quad (7)$$

$$z_{kk}^{hk} \in \{0, 1\} \quad k \in \mathcal{D}, h = 2, \dots, H \quad (8)$$

$$x_{ij} \in \{0, 1\} \quad (i, j) \in \mathcal{A} \quad (9)$$

As restrições (1) garantem que um e apenas um arco chega a cada nodo (excepto para o nodo raiz). Para cada nodo $k \in \mathcal{D}$, as restrições (2) estabelecem que o arco $(0, i)$ está no caminho para o nodo k na posição 1 se e só se existe um arco deixando o nodo i no caminho para o nodo k na posição 2. As restrições (3) garantem que, no caminho para o nodo k , um arco entra no nodo i na posição h se e só se um arco deixa o nodo i na posição $h + 1$. As restrições (4) estabelecem que, no caminho para o nodo k , um e apenas um arco entra no nodo k na posição H . As restrições (5) são as restrições de ligação que estabelecem a relação entre as variáveis x_{ij} e z_{ij}^{hk} e garantem, para cada k , que o arco (i, j) é incluído numa posição, e apenas numa, do caminho mais curto entre o nodo raiz e o nodo k se esse arco é incluído na solução. As restrições (6), (7), (8) e (9) são as restrições de integralidade das variáveis z_{ij}^{hk} e x_{ij} .

Para definir a relaxação linear deste problema, que designaremos por $HMCF_L$, substituímos as restrições de integralidade (6), (7), (8) e (9), respectivamente, por:

$$0 \leq z_{0j}^{1k} \leq 1 \quad j, k \in \mathcal{D}$$

$$0 \leq z_{ij}^{hk} \leq 1 \quad (i, j) \in \mathcal{A}, k \in \mathcal{D}, h = 2, \dots, H$$

$$0 \leq z_{kk}^{hk} \leq 1 \quad k \in \mathcal{D}, h = 2, \dots, H$$

$$0 \leq x_{ij} \leq 1 \quad (i, j) \in \mathcal{A}$$

Para cada $k \in \mathcal{D}$ o sistema de restrições (2), (3), (4), (6), (7) e (8) constitui uma formulação compacta para o problema do caminho mais curto com restrições de salto entre o nodo 0 e o nodo k . Notamos que, para cada k , cada um destes sistemas de equações usam as variáveis z_{kk}^{hk} , $h = 2, \dots, H$, e é precisamente por isto que cada um destes sistemas permite a existência de caminhos entre o nodo raiz e o nodo k com menos de H saltos. Quando um caminho contém menos de H arcos são necessários

lacetes no nodo k para obter uma solução contendo exactamente H variáveis com valor igual a 1.

4.3 Uma relaxação lagrangeana

Para se obter uma relaxação lagrangeana para a formulação $HMCF$, associam-se os multiplicadores de Lagrange w_i^{1k} , para todo $i, k \in \mathcal{D}$, às restrições de conservação de fluxo (2), os multiplicadores de Lagrange w_i^{hk} , para todo $i, k \in \mathcal{D}$ e $h = 2, \dots, H-1$, às restrições de conservação de fluxo (3) e os multiplicadores de Lagrange w_k^{Hk} , para todo $k \in \mathcal{D}$, às restrições de conservação de fluxo (4), e dualizamos estas restrições de conservação de fluxo de modo lagrangeano. Obtemos o seguinte problema que designamos por $HMCF_w$.

$$\begin{aligned} \min \quad & \sum_{(i,j) \in \mathcal{A}} c_{ij} x_{ij} - \sum_{i,k \in \mathcal{D}} w_i^{1k} m_i^{1k} - \sum_{i,k \in \mathcal{D}} \sum_{h=2}^{H-1} w_i^{hk} m_i^{hk} - \sum_{k \in \mathcal{D}} w_k^{Hk} m_k^{Hk} \\ \text{sujeito a} \quad & \\ & (1), (5), (6), (7), (8), (9) \end{aligned}$$

em que

$$\begin{aligned} m_i^{1k} &= z_{0i}^{1k} - \sum_{j \in \mathcal{N}} z_{ij}^{2k}, & i, k \in \mathcal{D} \\ m_i^{hk} &= \sum_{j \in \mathcal{N}} z_{ji}^{hk} - \sum_{j \in \mathcal{V}} z_{ij}^{h+1,k}, & i, k \in \mathcal{D}, h = 2, \dots, H-1 \\ m_k^{Hk} &= \sum_{j \in \mathcal{N}} z_{jk}^{Hk} - 1, & k \in \mathcal{D} \end{aligned}$$

Efectuando algumas simplificações à função objectivo obtemos o seguinte problema.

$$\begin{aligned} \min \quad & \sum_{(i,j) \in \mathcal{A}} c_{ij} x_{ij} - \sum_{i,k \in \mathcal{D}} w_i^{1k} z_{0i}^{1k} \\ & - \sum_{k \in \mathcal{D}} \sum_{h=2}^{H-1} \sum_{i,j \in \mathcal{D}} (w_j^{hk} - w_i^{h-1,k}) z_{ij}^{hk} \\ & - \sum_{k \in \mathcal{D}} \sum_{j \in \mathcal{D}} (w_k^{Hk} - w_j^{H-1,k}) z_{jk}^{Hk} + \sum_{k \in \mathcal{D}} w_k^{Hk} \end{aligned}$$

Fazendo

$$\eta_{0i}^{1k} = -w_i^{1k}, \quad \text{para } i, k \in \mathcal{D}$$

$$\eta_{ij}^{hk} = w_i^{h-1,k} - w_j^{hk}, \quad \text{para } i, j, k \in \mathcal{D}, h = 2, \dots, H-1$$

$$\eta_{jk}^{Hk} = w_j^{H-1,k} - w_k^{Hk}, \quad \text{para } j, k \in \mathcal{D}$$

e reescrevendo a função objectivo, obtemos:

$$\min \sum_{(i,j) \in \mathcal{A}} c_{ij} x_{ij} + \sum_{i,k \in \mathcal{D}} \eta_{0i}^{1k} z_{0i}^{1k} + \sum_{i,j,k \in \mathcal{D}} \sum_{h=2}^{H-1} \eta_{ij}^{hk} z_{ij}^{hk} + \sum_{j,k \in \mathcal{D}} \eta_{jk}^{Hk} z_{jk}^{Hk} + \sum_{k \in \mathcal{D}} w_k^{Hk}$$

Obtemos, deste modo, o seguinte problema relaxado:

$$\min \sum_{(i,j) \in \mathcal{A}} c_{ij} x_{ij} + \sum_{(i,j) \in \mathcal{A}} \sum_{k \in \mathcal{D}} \sum_{h=1}^H \eta_{ij}^{hk} z_{ij}^{hk} + \sum_{k \in \mathcal{D}} \sum_{h=2}^H \eta_{kk}^{hk} z_{kk}^{hk} + \sum_{k \in \mathcal{D}} w_k^{Hk}$$

sujeito a

$$\begin{aligned} \sum_{i \in \mathcal{N}} x_{ij} &= 1, \quad j \in \mathcal{D} \\ \sum_{h=1}^H z_{ij}^{hk} &\leq x_{ij}, \quad (i,j) \in \mathcal{A}, k \in \mathcal{D} \\ z_{0j}^{1k} &\in \{0, 1\} \quad j, k \in \mathcal{D} \\ z_{ij}^{hk} &\in \{0, 1\} \quad (i,j) \in \mathcal{A}, k \in \mathcal{D}, i \neq 0, k, h = 2, \dots, H \\ z_{kk}^{hk} &\in \{0, 1\} \quad k \in \mathcal{D}, h = 2, \dots, H \\ x_{ij} &\in \{0, 1\} \quad (i,j) \in \mathcal{A} \end{aligned}$$

Esta relaxação envolve $\mathcal{O}(n^2 H)$ multiplicadores, $\mathcal{O}(n|\mathcal{A}|H)$ variáveis e $\mathcal{O}(n|\mathcal{A}|)$ restrições.

Para um dado conjunto de multiplicadores de Lagrange w_j^{hk} , com $j, k \in \mathcal{D}$, $h = 1, \dots, H$, é possível transformar a relaxação lagrangeana de $HMCF$ num problema envolvendo apenas as variáveis x_{ij} e as variáveis z_{kk}^{hk} e uma sua solução pode ser obtida através da resolução de um problema de minimização simples e de dois problemas de inspecção. Vejamos como.

Seja x_{ij}^* o valor óptimo para as variáveis x_{ij} , $(i,j) \in \mathcal{A}$. Usando as restrições $\sum_{h=1}^H z_{ij}^{hk} \leq x_{ij}$, $(i,j) \in \mathcal{A}, k \in \mathcal{D}$, e as restrições de integralidade nas variáveis z_{ij}^{hk} , no

máximo, uma das variáveis z_{ij}^{hk} , $h = 1, \dots, H$, pode ter valor igual a 1 e todas as outras variáveis terão valor igual a 0. Tome-se $\gamma_{ij}^k = \min_{h=2, \dots, H} \{\eta_{ij}^{hk}\}$ e seja p o índice para o qual este valor mínimo ocorre. Portanto, se $\gamma_{ij}^k \geq 0$ tem-se $z_{ij}^{hk} = 0$ para $h = 2, \dots, H$. Caso contrário, se $\gamma_{ij}^k < 0$ tem-se $z_{ij}^{pk} = x_{ij}^*$ e $z_{ij}^{hk} = 0$ para $h = 2, \dots, H$ e $h \neq p$. Consequentemente,

$$\sum_{h=2}^H \eta_{ij}^{hk} z_{ij}^{hk} = \min\{0, \gamma_{ij}^k\} z_{ij}^{pk} = \min\{0, \gamma_{ij}^k\} x_{ij}^*.$$

De modo semelhante se obtém

$$\eta_{0j}^{1k} z_{0j}^{1k} = \min\{0, \gamma_{0j}^k\} x_{0j}^*, \text{ com } \gamma_{0j}^k = \eta_{0j}^{1k}.$$

Portanto, as restrições de integralidade nas variáveis z_{ij}^{hk} da relaxação lagrangeana podem ser removidas da formulação *HMCF* e esta pode ser reescrita do seguinte modo:

$$\min \sum_{(i,j) \in \mathcal{A}} \left(c_{ij} + \sum_{k \in \mathcal{D}} \min\{0, \gamma_{ij}^k\} \right) x_{ij} + \sum_{k \in \mathcal{D}} \sum_{h=2}^H \eta_{kk}^{hk} z_{kk}^{hk} + \sum_{k \in \mathcal{D}} w_k^{Hk}$$

sujeito a

$$\sum_{i \in \mathcal{N}} x_{ij} = 1, \quad j \in \mathcal{D}$$

$$z_{kk}^{hk} \in \{0, 1\}, \quad k \in \mathcal{D}, h = 2, \dots, H$$

$$x_{ij} \in \{0, 1\}, \quad (i, j) \in \mathcal{A}$$

Notamos, agora, que esta formulação modificada pode ser separada em dois sub-problemas de inspecção bastante simples. Um que envolve apenas as variáveis x_{ij} , $(i, j) \in \mathcal{A}$, e outro que envolve apenas as variáveis z_{kk}^{hk} , $k \in \mathcal{D}$, $h = 2, \dots, H$.

Uma vez definida esta formulação modificada da relaxação lagrangeana há que obter uma aproximação para o valor dos multiplicadores e para tal serão aplicados os métodos descritos no Capítulo 3.

Notamos que esta relaxação satisfaz a propriedade da integralidade, porque os sub-problemas resultantes na relaxação HMCF_w têm soluções inteiras, o que significa que o

melhor valor teórico associado a esta relaxação é o mesmo que o associado à relaxação linear $HMCF_L$. Contudo, em cada iteração dos métodos que aproximam o valor desta relaxação é apenas necessário resolver problemas muito simples, pelo que esta relaxação parece ser bastante eficiente.

Quando os multiplicadores são inicializados a zero, o limite inferior obtido na primeira iteração corresponde ao valor da arborescência mínima que, de um modo geral, é melhor que o valor óptimo do correspondente subproblema de inspecção. O problema da arborescência mínima pode ser resolvido eficientemente usando o algoritmo descrito por Fischetti e Toth [10]. Além disso, resolver este problema em vez de um subproblema de inspecção permite usar em cada iteração dos métodos de optimização descritos anteriormente a heurística descrita em Gouveia [15] para obter um limite superior do valor óptimo do problema. Por estas razões em vez da resolução do subproblema nas variáveis x por inspecção, determina-se o valor destas variáveis x utilizando o algoritmo de Fischetti e Toth

4.4 Aplicação dos métodos ao problema

Nesta secção descreveremos os elementos da relaxação lagrangeana da formulação $HMCF$ necessários à implementação dos métodos descritos no Capítulo 3.

Em cada iteração dos algoritmos temos de determinar $w_{t+1} = w_t + p_t d_t$. O valor inicial para os multiplicadores w_t é a solução nula, isto é $w_t = 0$.

No Passo 1 de todos os algoritmos, dado um valor para os multiplicadores w , temos de obter a solução óptima da função dual lagrangeana $\theta(w)$ da relaxação lagrangeana de $HMCF$ que é dada por

$$\theta(w) = \min \sum_{(i,j) \in A} \left(c_{ij} + \sum_{k \in \mathcal{D}} \min\{0, \gamma_{ij}^k\} \right) x_{ij} + \sum_{k \in \mathcal{D}} \sum_{h=2}^H \eta_{kk}^{hk} z_{kk}^{hk} + \sum_{k \in \mathcal{D}} w_k^{Hk}$$

sujeito a

$$\sum_{i \in \mathcal{N}} x_{ij} = 1, \quad j \in \mathcal{D}$$

$$z_{kk}^{hk} \in \{0, 1\}, \quad k \in \mathcal{D}, h = 2, \dots, H$$

$$x_{ij} \in \{0, 1\}, \quad (i, j) \in \mathcal{A}$$

44CAPÍTULO 4. O PROBLEMA HMST E UMA SUA RELAXAÇÃO LAGRANGEANA

sendo $w = (w_i^{1k}, w_i^{hk}, w_k^{Hk})$, com $i, k \in \mathcal{D}$, $h = 2, \dots, H-1$, o vector dos multiplicadores de lagrange, $\eta_{kk}^{hk} = w_k^{h-1,k} - w_k^{hk}$, para $k \in \mathcal{D}$ e $h = 2, \dots, H-1$, e γ_{ij}^k descrito na secção anterior como sendo $\gamma_{0j}^k = \eta_{0j}^{1k} = -w_i^{1k}$ para $j, k \in \mathcal{D}$ e $\gamma_{ij}^k = \min_{h=2, \dots, H} \eta_{ij}^{hk}$ com $\eta_{ij}^{hk} = w_i^{h-1,k} - w_j^{hk}$ para $(i, j) \in \mathcal{A}$, $k \in \mathcal{D}$ e $h = 2, \dots, H$. Na secção anterior descreveu-se já a forma como se obteria a sua solução óptima através de dois subproblemas de inspecção, um envolvendo apenas as variáveis x e outro apenas as variáveis z . Portanto, após determinar para cada $(i, j) \in \mathcal{A}$ os valores $\bar{c}_{ij} = c_{ij} + \sum_{k \in \mathcal{D}} \min\{0, \gamma_{ij}^k\}$ podemos em vez de resolver o problema de inspecção nas variáveis x obtendo para cada $j \in \mathcal{D}$ o $\min_{i \in \mathcal{D}} \{c_{ij} + \sum_{k \in \mathcal{D}} \min\{0, \gamma_{ij}^k\}\}$ sendo i^* o índice onde o mínimo é obtido e fazendo $x_{i^*j} = 1$ e $x_{ij} = 0$ para $i \neq i^*$, determinamos a arborescência mínima usando os custos modificados \bar{c}_{ij} , tal como referido, e usando o algoritmo de Fischetti e Toth.

Por inspecção obtemos o valor para as variáveis z fazendo $z_{kk}^{hk} = 1$ se $\eta_{kk}^{hk} \leq 0$ e $z_{kk}^{hk} = 0$ se $\eta_{kk}^{hk} > 0$, sendo $\eta_{kk}^{hk} = w_k^{h-1,k} - w_k^{hk}$ para $k \in \mathcal{D}$ e $h = 2, \dots, H$. Facilmente se obtém agora o valor $\theta(w)$.

O vector subgradiente g é dado por

$$g = (m_i^{1k}, m_i^{hk}, m_i^{Hk}) = \left(z_{0i}^{1k} - \sum_{j \in \mathcal{N}} z_{ij}^{2k}, \sum_{j \in \mathcal{N}} z_{ji}^{hk} - \sum_{j \in \mathcal{N}} z_{ij}^{h+1,k}, \sum_{j \in \mathcal{N} \setminus \{k\}} z_{jk}^{Hk} + z_{kk}^{Hk} - 1 \right).$$

Sendo $w_t = (w_i^{1k}, w_i^{hk}, w_k^{Hk})_t$ o valor dos multiplicadores obtido na iteração t , temos x_t e z_t o valor das variáveis obtido nessa iteração e portanto $\theta(w_t) = (w_i^{1k}, w_i^{hk}, w_k^{Hk})_t$ será o valor da função dual e $g_t = (m_i^{1k}, m_i^{hk}, m_i^{Hk})_t$ o valor do subgradiente nessa iteração.

Agora no Passo 2 teremos de calcular a direcção d_t e o tamanho de passo p_t . É neste passo que os métodos descritos são distintos.

No método subgradiente torna-se necessário calcular o quadrado da norma do subgradiente $\|g_t\|^2$ que é dado por

$$\sum_{i, k \in \mathcal{D}} (m_i^{1k})^2 + \sum_{(i, j) \in \mathcal{A}} \sum_{k \in \mathcal{D}} (m_i^{hk})^2 + \sum_{k \in \mathcal{D}} (m_i^{Hk})^2$$

e que é usado no cálculo do tamanho do passo na Regra *Held, Wolf & Crowder*.

Para o cálculo da direcção pelo método subgradiente radar é necessário determinar o declive do plano de suporte dado por

$$g_s g_t = \begin{bmatrix} m_i^{1k} & m_i^{hk} & m_i^{Hk} \end{bmatrix} \begin{bmatrix} m_i^{1k} \\ m_i^{hk} \\ m_i^{Hk} \end{bmatrix} = m_i^{1k} \cdot m_i^{1k} + \sum_h m_i^{hk} m_i^{hk} + m_i^{Hk} m_i^{Hk}$$

Para aplicar o método do feixe de subgradientes ao problema $HMCF_w$ é necessário determinar a solução óptima do problema dual quadrático para se seleccionar a direcção d com tamanho de passo constante e igual a 1, dada por

$$\bar{\theta}_{\mathcal{B},d}(d) = \max_d \{ \theta_{\mathcal{B},d}(d) - \frac{1}{2r_t} \|d\|^2 \},$$

sendo $\theta_{\mathcal{B},d}(d) = \min_{k \in \mathcal{B}} \{ \beta_k + g_k d \} + \theta(w_t)$, β_k o erro de linearização e o parâmetro da região de confiança, $r_t > 0$, interpretado como o tamanho de passo. É necessário, também, verificar se $\theta(w_t + d) - \theta(w_t) \geq tol$ para se efectuar um *passo autêntico* ($d_t = d$), caso contrário efectua-se um *passo nulo* ($d_t = 0$).

Capítulo 5

Resultados computacionais

Neste capítulo apresentamos alguns resultados computacionais obtidos pelos métodos descritos no Capítulo 3 usando instâncias do problema HMST. Os resultados são comparados quanto ao número de iterações, ao tempo de execução e à qualidade do limite inferior obtido. Os melhores resultados que se esperam são os da relaxação linear uma vez que esta relaxação lagrangiana goza da propriedade da integralidade.

Com o objectivo de obter resultados computacionais para comparar o desempenho do método subgradiente e do método subgradiente radar usámos instâncias do problema HMST com $n = 20, 40$ e 60 nodos e, para cada valor de n , um conjunto de três grupos de instâncias. Dois destes grupos correspondem a instâncias euclidianas e o terceiro corresponde a instâncias aleatórias.

Para se obterem as instâncias euclidianas, primeiro foram geradas aleatoriamente as coordenadas de n pontos numa grelha de dimensão 100×100 e, posteriormente, o custo, c_{ij} , de cada aresta $\{i, j\}$ foi obtido tomando apenas a parte inteira da distância euclidiana entre os pontos i e j gerados na rede. Para as distâncias euclidianas foram consideradas duas localizações para o nodo raiz, uma em que a raiz se encontra no centro da grelha e outra em que a raiz se encontra num extremo da grelha. Estas instâncias foram designadas, respectivamente, por TC e TE. O custo c_{ij} , de cada aresta $\{i, j\}$ das instâncias aleatórias foi obtido utilizando uma distribuição uniforme no intervalo $[0, 100]$. Para estas instâncias considerou-se que o nodo raiz era o nodo 0 e foram denotadas por TR. Os detalhes da geração de todas estas instâncias podem ser consultadas em

[8].

Para o valor $n = 20$ com distâncias euclidianas foram gerados cinco grafos completos com a raiz no centro correspondendo a cinco problemas teste TC que designamos por TC1, TC2, TC3, TC4 e TC5, foram gerados cinco grafos completos com raiz num extremo correspondendo a cinco problemas testes TE que denotamos por TE1, TE2, TE3, TE4 e TE5 e, com distâncias aleatórias foram gerados cinco grafos completos correspondendo a cinco problemas teste TR que designamos por TR1, TR2, TR3, TR4 e TR5.

Para o valor $n = 40$ com distâncias euclidianas foram gerados dois grafos completos com a raiz no centro correspondendo a dois problemas teste TC que designamos por TC1 e TC2, foram gerados dois grafos completos com raiz num extremo correspondendo a dois problemas testes TE que denotamos por TE1 e TE2 e, com distâncias aleatórias foram gerados dois grafos completos correspondendo a dois problemas teste TR que designamos por TR1 e TR2.

Para o valor $n = 60$ com distâncias euclidianas foi gerado um grafo completo com a raiz no centro correspondendo a um problema teste TC que designamos por TC1, foi gerado um grafo completo com raiz num extremo correspondendo a um problema teste TE que denotamos por TE1 e, com distâncias aleatórias foi gerado um grafo completo correspondendo a um problema teste TR que designamos por TR1.

Para cada valor de n e para cada um destes testes testámos os valores de 3, 4, 5 para o parâmetro H .

Uma desvantagem da relaxação lagrangeana HMC_w , apresentada no Capítulo 4, encontra-se no facto de envolver um elevado número de variáveis e restrições. Na resolução do problema, este facto traduz-se num enorme consumo de tempo computacional visto os modelos serem bastante grandes. O número de arcos em cada instância para cada valor de n é igual a $n + n \times (n - 1)$, portanto, para cada valor de $n = 20, 40$ e 60 têm-se, respectivamente, 400, 1600 e 3600 arcos. De forma a reduzir o tamanho dos problemas teste foi usado o seguinte teste de eliminação de arcos das instâncias [15]:

se $c_{ij} > c_{0j}$, então nenhuma solução óptima possui o arco (i, j) . Se $c_{ij} = c_{0j}$ ($i \neq 0$), então existe uma solução óptima que não inclui o arco (i, j) .

Isto significa que se a solução encontrada possuir tais arcos é possível encontrar uma outra solução admissível com o mesmo custo (nos casos em que $c_{ij} = c_{0j}$) ou, então uma solução admissível melhor (nos casos em que $c_{ij} > c_{0j}$) se removermos o arco (i, j) e incluirmos o arco $(0, j)$. Deste modo, o arco (i, j) pode ser eliminado do problema, permitindo, como consequência, eliminar também algumas restrições.

Este teste de eliminação de arcos é aplicado em todas as instâncias antes de se resolver cada relaxação lagrangeana ou antes de se resolver a relaxação em programação linear, $HMCF_L$ (os valores obtidos por esta relaxação linear [8, 15] serão usados para comparação com os nossos resultados). Notamos que este teste é mais eficaz quando aplicado a instâncias TC do que a instâncias TE ou TR e é mais eficaz para instâncias TR do que para instâncias TE. Deste modo, pode-se concluir que as instâncias reduzidas TE são maiores do que as instâncias reduzidas TC ou TR, o que sugere que as instâncias TE são mais difíceis de resolver do que as outras instâncias.

Uma das desvantagens do uso de métodos de otimização por subgradiente é a necessidade do uso de parâmetros que dependem do processo usado para resolver o problema associado ao método e à instância teste. Nos métodos usados, os parâmetros que fizemos variar foram o tamanho do passo e o número máximo de iterações permitido.

Após terem sido testados vários valores para o tamanho do passo inicial, chegamos à conclusão que os valores para os quais obtínhamos melhores aproximações para o valor ótimo do problema, no sentido em que a diferença entre o limite inferior da relaxação linear e o limite inferior determinado pela relaxação lagrangeana, é menor são os indicados na Tabela 5.1 onde *HWC* denota a regra de *Held, Wolf & Crowder*, *C* denota a regra de *Crowder* e *CFM* denota a regra de *Camerini, Fratta, Maffioli modificada* do método subgradiente e *Radar* denota o método subgradiente radar. Os valores que utilizamos para fazer variar o número máximo de iterações foram 2000, 5000, 7000 e 9000.

Os métodos iterativos de otimização por subgradiente e por subgradiente radar descritos no Capítulo 3 foram implementados computacionalmente usando a linguagem de programação *FORTRAN*. Os resultados computacionais foram obtidos num *Intel(R) Core(TM)* com 2047 MB de *RAM*, com processador de 2.13 GHz e são apresentados nas

Tabela 5.1: Intervalo de valores usado para o tamanho do passo

	$N = 20$	$N = 40$	$N = 60$
<i>HWC</i>	$0.5 \leq p \leq 2.0$	$6.0 \leq p \leq 9.0$	$2.0 \leq p \leq 8.0$
<i>C</i>	$0.3 \leq p \leq 2.0$	$2.0 \leq p \leq 9.0$	$2.0 \leq p \leq 8.0$
<i>CFM</i>	$0.2 \leq p \leq 1.0$	$2.0 \leq p \leq 3.0$	$1.0 \leq p \leq 4.0$
<i>Radar</i>	$0.5 \leq p \leq 8.0$	$3.0 \leq p \leq 9.5$	$4.0 \leq p \leq 8.0$

secções seguintes.

5.1 Resultados obtidos pelo método subgradiente

Nesta secção são apresentados os resultados computacionais obtidos pelo método subgradiente em tabelas cuja primeira coluna, denotada por *Prob* e por $|V|$, identifica o problema e o número de nodos; a segunda coluna indica o valor do parâmetro H considerado; a terceira e quarta coluna, denotadas por *LimSup* que identifica o limite superior (obtido por uma heurística, consultar [15] para detalhes) e *LimLin* que identifica inferior obtido pela relaxação linear HMCF_L ; as duas colunas seguintes referem-se aos resultados obtidos pela relaxação lagrangeana quando usada a direcção *Held, Wolf & Crowder* denotada por *HWC*, a primeira destas duas colunas, denotada por *valor* e por *Gap* (em baixo), mostra o melhor limite inferior e o correspondente valor do *gap* definido por $\text{gap} = \frac{\text{LimSup} - \text{valor}}{\text{LimSup}} \times 100$ e a segunda destas duas colunas, denotada por *It-eração* e por *tempo* (em baixo), mostra o número de iterações usadas pelo procedimento de optimização por subgradiente indicado para obter o melhor limite inferior registado na coluna anterior e o tempo computacional, em segundos, usado para obter tal valor; as duas colunas seguintes identificam os resultados obtidos usando a direcção de *Crowder* denotada por *C* e de modo análogo para as duas últimas colunas seguintes que identificam os resultados obtidos pela direcção *Camerini, Fratta, Maffioli modificada* denotada por *CFM*.

Notamos que quando o $\text{LimSup} = \text{LimLin}$ este valor corresponde ao valor da solução

ótima. Consequentemente, sempre que se fizer referência ao valor ótimo de uma certa instância, este coincide com o valor *LimLin*.

Para obtermos os resultados computacionais, começámos por executar o método subgradiente com direcção *Held, Wolfe & Crowder* para todos os testes já referidos. Os primeiros tamanhos de passo considerados foram 0.5, 1.0, 1.5 e 2.0 com número máximo de iterações para TC e TR usando 2000 e para intâncias TE usando 5000. Procedemos de modo análogo para as direcções de *Crowder* e de *Camerini, Fratta, Maffioli modificada*. Os melhores resultados obtidos foram tabelados para que a sua análise fosse mais fácil e mais perceptível. A partir dessas tabelas foram identificados todos os casos em que o método subgradiente não alcançou o *LimLin* (limite inferior da relaxação linear). De seguida, iniciámos o processo de melhoramento destes resultados para que o valor obtido fosse o melhor possível usando o menor número de iterações e o menor tempo de execução. Para tal, foi necessário testarmos diversos valores para o tamanho do passo com o número máximo de iterações.

Observando os resultados registados foi possível identificar mais facilmente quais os tamanhos de passo e número de iterações para os quais o melhor valor da relaxação lagrangeana era alcançado. Ou seja, se o melhor valor alcançado até ao momento usasse tamanhos de passos compreendidos entre 0.5 e 1.5, fazíamos variar o tamanho do passo nesse intervalo, executando novamente o algoritmo e analisando as possíveis alterações. Se essas alterações melhorassem os valores anteriormente obtidos eram registadas e sucessivamente alterados até serem alcançadas as melhores combinações de valores possíveis. Outra estratégia utilizada foi alterar o número máximo de iterações e procedemos de modo análogo à estratégia anterior. Isto é, suponhamos que o valor alcançado não estava muito próximo do valor da relaxação linear, então observávamos o número de iterações que foram necessárias para alcançar esse valor. Se o número de iterações usadas estivesse próximo do número de iterações limite atribuído inicialmente, alterávamos esse valor máximo e executávamos novamente. Por exemplo, suponhamos que atribuíamos inicialmente como número máximo de iterações o valor 2000, se depois de executarmos verificássemos que o valor alcançado não era o mais desejado e que o número de iterações que foram necessárias está muito próximo de 2000, na expecta-

tiva de melhorarmos esse resultado, mudávamos esse número de iterações para 5000, executávamos novamente e registávamos as possíveis alterações.

Muitas destas modificações dos valores do tamanho do passo e do número máximo de iterações iniciais permitiram melhorar os resultados computacionais de modo satisfatório visto que a maioria alcançou o *LimLin*, mas mesmo que não fosse alcançado conseguíamos uma melhor combinação de valores quer obtendo um melhor valor da relaxação lagrangeana, quer usando um menor número de iterações ou usando menos tempo de execução. No entanto, por vezes, só a alteração de um desses dados iniciais não era o suficiente e por isso conjugámos ambas as estratégias.

Estes procedimentos foram aplicados às três direcções para todas as instâncias e os melhores resultados obtidos podem ser observados nas Tabelas 5.2, 5.3 e 5.4. A primeira tabela refere-se aos resultados obtidos para os testes TC, a segunda tabela aos resultados para os testes TE e a terceira aos resultados para os testes TR.

Por observação da Tabela 5.2 podemos concluir que o valor *LimLin* não foi alcançado para alguns testes cujas instâncias têm um maior número de nodos, como é o caso de TC1_40 ($H = 5$), TC2_40 ($H = 4$) e TC1_60 ($H = 4$) para as três regras e não foi alcançado para TC1_60 ($H = 3$) para as regras *HWC* e *C* e para TC1_60 ($H = 5$) para as regras *HWC* e *CFM*.

Apesar de, para $H = 3$, a maioria dos testes alcançarem o valor de *LimLin* ou o valor óptimo, o valor do correspondente *gap* é elevado revelando que estes limites são de fraca qualidade contudo são os melhores que se conseguem obter, uma vez que coincidem com o *LimLin*, tal como acontece para TC5_20 ($H = 4$), TC1_40 ($H = 4$), TC2_40 ($H = 5$).

Os tempos de execução não são muito elevados excepto para aqueles testes que não alcançaram o *LimLin* como é o caso, por exemplo, de TC1_60 ($H = 5$) em que o tempo de execução excede os 150 segundos nas três regras, sendo o maior tempo de execução de 294,78 segundos para a regra *C*.

O número de iterações necessárias para se alcançar o melhor valor possível para as instâncias com 20 nodos para as regras de *HWC* e de *C* não excede as 2000 iterações enquanto que para a regra *CFM* para TC1_20 ($H = 4$) já se aproxima das 5000. Com

Tabela 5.2: Resultados computacionais para os testes TC.

$Prob, V $	H	$LimSup$	$LimLin$	<i>RegraHWC</i>		<i>RegraC</i>		<i>RegraCFM</i>	
				<i>Valor</i> (<i>Gap</i>)	<i>Iteração</i> (<i>Tempo</i>)	<i>Valor</i> (<i>Gap</i>)	<i>Iteração</i> (<i>Tempo</i>)	<i>Valor</i> (<i>Gap</i>)	<i>Iteração</i> (<i>Tempo</i>)
TC1, 20	3	340	339	339 (0.29)	779 (0.49)	339 (0.29)	748 (0.52)	339 (0.29)	850 (0.69)
	4	318	318	318 (0.00)	799 (0.69)	318 (0.00)	599 (0.61)	318 (0.00)	4199 (4.96)
	5	312	312	312 (0.00)	801 (0.91)	312 (0.00)	498 (0.72)	312 (0.00)	699 (1.17)
TC2, 20	3	365	364	364 (0.27)	697 (0.43)	364 (0.27)	599 (0.45)	364 (0.27)	2999 (2.53)
	4	338	338	338 (0.00)	150 (0.13)	338 (0.00)	187 (0.17)	338 (0.00)	183 (0.22)
	5	332	332	332 (0.00)	500 (0.56)	332 (0.00)	350 (0.48)	332 (0.00)	643 (1.03)
TC3, 20	3	343	339	339 (1.17)	797 (0.48)	339 (1.17)	849 (0.55)	339 (1.17)	2101 (1.73)
	4	306	306	306 (0.00)	701 (0.62)	306 (0.00)	750 (0.70)	306 (0.00)	691 (0.80)
	5	296	296	296 (0.00)	550 (0.62)	296 (0.00)	449 (0.54)	296 (0.00)	500 (0.81)
TC4, 20	3	390	387	387 (0.77)	196 (0.11)	387 (0.77)	234 (0.14)	387 (0.77)	699 (0.50)
	4	376	376	376 (0.00)	1649 (1.36)	376 (0.00)	950 (0.82)	376 (0.00)	2200 (2.51)
	5	364	364	364 (0.00)	1100 (1.21)	364 (0.00)	751 (0.83)	364 (0.00)	1949 (2.92)
TC5, 20	3	347	347	347 (0.00)	242 (0.16)	347 (0.00)	198 (0.13)	347 (0.00)	150 (0.11)
	4	326	324	324 (0.61)	650 (0.53)	324 (0.61)	550 (0.47)	324 (0.61)	649 (0.82)
	5	310	310	310 (0.00)	1 (0.00)	310 (0.00)	1 (0.00)	310 (0.00)	1 (0.00)
TC1, 40	3	609	605	605 (0.66)	1784 (8.25)	605 (0.66)	2648 (12.15)	605 (0.66)	1600 (13.03)
	4	548	547	547 (0.18)	3493 (26.11)	547 (0.18)	2193 (16.81)	547 (0.18)	2591 (33.22)
	5	522	522	521 (0.19)	5739 (60.73)	521 (0.19)	1947 (21.70)	521 (0.19)	4896 (87.50)
TC2, 40	3	566	566	566 (0.00)	650 (3.18)	566 (0.00)	494 (2.58)	566 (0.00)	685 (5.66)
	4	519	518	517 (0.39)	5400 (40.50)	517 (0.39)	3441 (28.27)	517 (0.39)	4350 (52.68)
	5	496	494	494 (0.40)	4138 (43.80)	494 (0.40)	2200 (23.65)	494 (0.40)	5986 (102.98)
TC1, 60	3	866	862	861 (0.58)	2098 (37.53)	861 (0.58)	3246 (58.90)	862 (0.46)	4848 (106.22)
	4	781	778	777 (0.51)	6142 (171.64)	777 (0.51)	5594 (161.11)	776 (0.64)	3843 (132.43)
	5	734	734	732 (0.27)	4292 (170.85)	734 (0.00)	6200 (294.78)	733 (0.14)	3200 (154.92)

o aumento do número de nodos, também o número de iterações necessárias para se alcançar o melhor valor possível aumenta, aproximando-se das 6000 para 40 nodos e

das 7000 iterações para 60 nodos.

Estes resultados revelam que, com o aumento do número de nodos, torna-se mais difícil obter bons resultados em menor tempo e em menor número de iterações sendo na maioria dos casos, a regra de *Camerini, Fratta, Maffioli* a menos eficaz para as instâncias TC.

As instâncias TE são as que obtêm piores resultados pelo método subgradiente como podemos observar na Tabela 5.3, visto que o valor *LimLin* só foi alcançado para TE2_20 ($H = 3$), TE3_20 ($H = 3$ e 5) para as três regras e para TE4_20 ($H = 3$) para a regra *CFM*. Os valores do *gap* para estes testes foram nulos excepto para TE4_20 ($H = 3$), deste modo podemos concluir que para as três regras os limites são de grande qualidade enquanto que para o caso do teste TE4_20 ($H = 3$) o valor que obtemos com a regra de *CFM* é um limite de fraca qualidade, embora o melhor que é possível obter com esta relaxação lagrangeana porque é igual a *LimLin*.

Os tempos de execução e o número de iterações utilizadas para os testes TE aumentam consoante o número de nodos para cada teste e mais uma vez as instâncias com 60 nodos são as mais demoradas alcançando os 520,79 segundos para TE1_60 ($H = 5$) usando a regra *C*. Quanto ao número de iterações, para estas instâncias é difícil estabelecer relações uma vez que um teste com 20 nodos pode utilizar no máximo 2000 iterações e outro teste com o mesmo número de nodos já pode necessitar de mais de 6000.

As instâncias TE necessitam de muitas iterações para alcançarem um valor aproximado a *LimLin*, como é o caso de TE1_40 ($H = 5$) com a regra *HWC* que atinge as 8798 iterações para só alcançar um valor aproximado a *LimLin*. Mais uma vez, tal como acontece com as instâncias TC, os maiores tempos de execução usados encontram-se para os testes com 60 nodos e as três regras necessitam de um elevado número de iterações.

Por esta análise podemos concluir que as instâncias TE, cujo nodo raiz se encontra num extremo da grelha de cada teste, são bastante difíceis e necessitam de processadores com bastante memória para que, pelo menos, o tempo de execução possa diminuir.

Contrariamente às instâncias TE, as instâncias TR são bastante fáceis como pode-

Tabela 5.3: Resultados computacionais para os testes TE.

$Prob, V $	H	$LimSup$	$LimLin$	<i>RegraHWC</i>		<i>RegraC</i>		<i>RegraCFM</i>	
				<i>Valor</i> (<i>Gap</i>)	<i>Iteração</i> (<i>Tempo</i>)	<i>Valor</i> (<i>Gap</i>)	<i>Iteração</i> (<i>Tempo</i>)	<i>Valor</i> (<i>Gap</i>)	<i>Iteração</i> (<i>Tempo</i>)
TE1, 20	3	449	444	443 (1.34)	3751 (3.27)	442 (1.56)	3100 (2.98)	443 (1.34)	4349 (4.78)
	4	385	385	384 (0.26)	3901 (5.29)	384 (0.26)	3751 (5.92)	384 (0.26)	3798 (6.55)
	5	366	364	361 (1.37)	4950 (8.58)	362 (1.09)	6549 (13.21)	362 (1.09)	8751 (19.81)
TE2, 20	3	435	435	435 (0.00)	347 (0.28)	435 (0.00)	348 (0.31)	435 (0.00)	351 (0.36)
	4	404	401	400 (0.99)	5701 (6.90)	400 (0.99)	3249 (4.64)	399 (1.24)	2099 (3.31)
	5	383	381	379 (1.04)	5301 (8.53)	379 (1.04)	3751 (7.11)	379 (1.04)	4150 (8.74)
TE3, 20	3	435	435	435 (0.00)	351 (0.30)	435 (0.00)	449 (0.41)	435 (0.00)	586 (1.58)
	4	396	395	394 (0.51)	1200 (1.45)	394 (0.51)	950 (1.21)	395 (0.25)	6401 (9.59)
	5	372	372	372 (0.00)	2449 (3.89)	372 (0.00)	1900 (3.07)	372 (0.00)	1601 (3.04)
TE4, 20	3	448	444	443 (1.12)	901 (0.71)	443 (1.12)	900 (0.84)	444 (0.89)	2699 (2.72)
	4	402	400	398 (1.00)	4998 (6.25)	397 (1.24)	3501 (4.54)	397 (1.24)	2799 (4.20)
	5	382	377	376 (1.57)	4751 (7.80)	375 (1.83)	2199 (3.67)	375 (1.83)	2700 (5.39)
TE5, 20	3	428	428	427 (0.23)	949 (0.75)	427 (0.23)	950 (0.78)	427 (0.23)	951 (0.98)
	4	376	376	376 (0.23)	400 (0.47)	376 (0.00)	448 (0.56)	376 (0.00)	447 (0.70)
	5	354	354	354 (0.00)	1150 (1.76)	354 (0.00)	801 (1.25)	354 (0.00)	1649 (3.62)
TE1, 40	3	708	702	701 (0.99)	7294 (47.85)	700 (1.13)	5000 (33.40)	701 (0.99)	6698 (67.12)
	4	627	625	623 (0.64)	6650 (71.20)	623 (0.64)	7447 (82.57)	622 (0.80)	4748 (72.85)
	5	590	588	584 (1.02)	8798 (129.27)	583 (1.19)	7897 (121.06)	583 (1.19)	8245 (176.64)
TE2, 40	3	710	701	700 (1.41)	6449 (41.94)	700 (1.41)	5349 (35.49)	700 (1.41)	6894 (66.30)
	4	625	620	617 (1.28)	6190 (64.94)	617 (1.28)	6649 (71.21)	617 (1.28)	6395 (95.50)
	5	581	579	574 (1.20)	6341 (92.52)	574 (1.20)	8200 (123.29)	573 (1.38)	4896 (105.93)
TE1, 60	3	1525	1503	1500 (1.64)	6897 (174.19)	1499 (1.70)	7248 (216.12)	1498 (1.77)	4098 (117.47)
	4	1345	1315	1300 (3.35)	4950 (198.34)	1304 (3.05)	8395 (365.95)	1300 (3.35)	6245 (280.68)
	5	1238	1213	1193 (3.63)	6642 (389.75)	1194 (3.55)	6442 (520.79)	1189 (3.96)	5795 (381.12)

mos observar na Tabela 5.4 visto que o $LimLin$ é alcançado para todos os testes com gap de 0.00 o que significa que são limites de boa qualidade excepto para TR3.20 (H

Tabela 5.4: Resultados computacionais para os testes TR.

$Prob, V $	H	$LimSup$	$LimLin$	<i>RegraHWC</i>		<i>RegraC</i>		<i>RegraCFM</i>	
				<i>Valor</i> (<i>Gap</i>)	<i>Iteração</i> (<i>Tempo</i>)	<i>Valor</i> (<i>Gap</i>)	<i>Iteração</i> (<i>Tempo</i>)	<i>Valor</i> (<i>Gap</i>)	<i>Iteração</i> (<i>Tempo</i>)
TR1, 20	3	168	168	168 (0.00)	196 (0.14)	168 (0.00)	193 (0.14)	168 (0.00)	149 (0.14)
	4	146	146	146 (0.00)	297 (0.28)	146 (0.00)	236 (0.27)	146 (0.00)	394 (0.53)
	5	137	137	137 (0.00)	1 (0.00)	137 (0.00)	1 (0.00)	137 (0.00)	1 (0.00)
TR2, 20	3	201	201	201 (0.00)	143 (0.09)	201 (0.00)	147 (0.14)	201 (0.00)	149 (0.28)
	4	161	161	161 (0.00)	800 (0.82)	161 (0.00)	589 (0.69)	161 (0.00)	684 (1.03)
	5	140	140	140 (0.00)	343 (0.45)	140 (0.00)	284 (0.48)	140 (0.00)	197 (0.41)
TR3, 20	3	157	153	153 (2.55)	401 (0.27)	153 (2.55)	401 (0.28)	153 (2.55)	699 (0.57)
	4	130	130	130 (0.00)	184 (0.17)	130 (0.00)	136 (0.14)	130 (0.00)	183 (0.22)
	5	121	121	121 (0.00)	1 (0.00)	121 (0.00)	1 (0.00)	121 (0.00)	1 (0.00)
TR4, 20	3	183	183	183 (0.00)	297 (0.21)	183 (0.00)	249 (0.18)	183 (0.00)	238 (0.21)
	4	158	158	158 (0.00)	499 (0.53)	158 (0.00)	498 (0.54)	158 (0.00)	589 (0.79)
	5	142	142	142 (0.00)	341 (0.47)	142 (0.00)	247 (0.36)	142 (0.00)	297 (0.55)
TR5, 20	3	148	148	148 (0.00)	237 (0.19)	148 (0.00)	195 (0.14)	148 (0.00)	147 (0.14)
	4	121	121	121 (0.00)	2201 (2.40)	121 (0.00)	4800 (5.34)	121 (0.00)	1651 (2.46)
	5	110	110	110 (0.00)	2349 (3.38)	110 (0.00)	1799 (2.61)	110 (0.00)	1699 (3.50)
TR1, 40	3	176	176	176 (0.00)	842 (4.85)	176 (0.00)	531 (2.95)	176 (0.00)	523 (4.50)
	4	149	148	148 (0.67)	892 (8.20)	148 (0.67)	548 (5.05)	148 (0.67)	740 (10.14)
	5	139	139	139 (0.00)	1283 (15.94)	139 (0.00)	735 (9.11)	139 (0.00)	942 (17.16)
TR2, 40	3	219	219	219 (0.00)	1149 (6.42)	219 (0.00)	636 (3.42)	219 (0.00)	685 (6.38)
	4	176	176	176 (0.00)	898 (8.10)	176 (0.00)	383 (3.40)	176 (0.00)	599 (8.33)
	5	155	155	155 (0.00)	1731 (21.34)	155 (0.00)	397 (5.06)	155 (0.00)	1300 (24.64)
TR1, 60	3	274	274	274 (0.00)	787 (18.63)	274 (0.00)	442 (10.24)	274 (0.00)	480 (12.18)
	4	207	207	207 (0.00)	838 (30.00)	207 (0.00)	474 (17.16)	207 (0.00)	440 (17.92)
	5	189	188	188 (0.53)	1697 (84.80)	188 (0.53)	892 (45.10)	188 (0.53)	989 (57.36)

= 3) e TR1.60 ($H = 5$). O número de iterações não ultrapassa os 5000 sendo o número máximo de iterações de 4800 para TR5.20 ($H = 4$) para a regra C e os tempos de exe-

cução não são muito elevados. O tempo de execução mais elevado é de 84.80 segundos com a regra *HWC* para TR1_60 ($H = 5$). Deste modo, podemos concluir que, pelo método subgradiente, as instâncias TR são as mais fáceis e aquelas para as quais se alcançam os melhores resultados.

Nenhuma das três regras sobressai em relação às outras relativamente ao número de iterações porque se num teste uma das regras usa menos iterações, noutro teste já é outra regra que necessita de menos iterações. Contudo, em termos de tempo de execução podemos verificar por observação da Tabela 5.4 que a regra *C* é a mais rápida quando o número de nodos aumenta.

Através da análise das tabelas apresentadas nesta secção podemos verificar que não foi possível obter o *LimLin* para todos os testes sendo os TE para 20, 40 e 60 nodos e os TC para 60 nodos os mais difíceis.

Os melhores resultados para os testes TC e TR com 20 nodos e com a direcção de *Held, Wolf & Crowder* alcançaram o limite inferior *LimLin* com rapidez. Por outro lado, para a maioria dos testes TE o valor alcançado não atinge o valor do limite inferior para nenhum tamanho de passo e número de iterações usados, como já foi referido.

Os tempos de execução para os testes com 20 nodos variam entre 0.00 e 7.80 segundos. Quando o valor da relaxação lagrangeana não alcança o valor *LimLin* do limite inferior (testes TE), estes diferem no máximo em três unidades.

Para a direcção de *Crowder*, verifica-se novamente que os testes com 20 nodos com as condições de tamanho de passo e número máximo de iterações impostas, que não alcançam *LimLin* são os testes de instâncias TE e os tempos de execução dos mesmos variam entre 0.00 e 13.21 segundos. Mais uma vez, quando o valor da relaxação lagrangeana não atinge o valor do limite inferior, estes diferem em três unidades.

De modo análogo, se verifica a mesma situação para a direcção de *Camerini, Fratta, Maffioli modificada*. Os tempos de execução variam entre 0.00 e 19.81 segundos, e o valor da relaxação lagrangeana difere, no máximo, em três unidades do valor *LimLin* do limite inferior quando estes não são iguais.

Quando se comparam os resultados obtidos pelas três direcções pelo método subgradiente podemos verificar, que entre as três, para 20 nodos a que obtém piores resultados,

na maioria dos casos, é a direcção *Camerini, Fratta, Maffioli modificada* visto que alcança o melhor resultado com pior tempo de execução.

Quanto aos problemas de 40 nodos, para as três direcções, pode-se observar que os testes de rápida execução e que obtêm o *LimLin* são, novamente, para as instâncias TC e TR. Quando o valor da relaxação lagrangiana não é igual ao valor do limite inferior, estes diferem no máximo em seis unidades e os tempos de execução são mais extensos que os de 20 nodos.

Comparando os resultados entre as três direcções, para os testes de 40 nodos, podemos verificar que a que obtém piores resultados, na maioria dos casos, é novamente a de *Camerini, Fratta, Maffioli modificada* com o pior número de iterações. Os tempos de execução para 40 nodos variam entre 2.95 e 129.27 segundos. Quando não se alcança o óptimo ou o limite inferior da relaxação linear, a diferença entre o melhor valor obtido e o óptimo é de 6 unidades.

Para 60 nodos, os tempos de execução variam entre os 2.58 e os 520.79 segundos e a diferença entre o valor da relaxação lagrangeana e o valor *LimLin* do limite inferior da relaxação linear é de no máximo 24 unidades.

Por comparação entre as três direcções para o método subgradiente, a direcção que apresenta melhores resultados na maioria dos casos quer pelo tempo de execução e número de iterações é a de *Crowder* revelando-se a direcção mais eficaz entre as três.

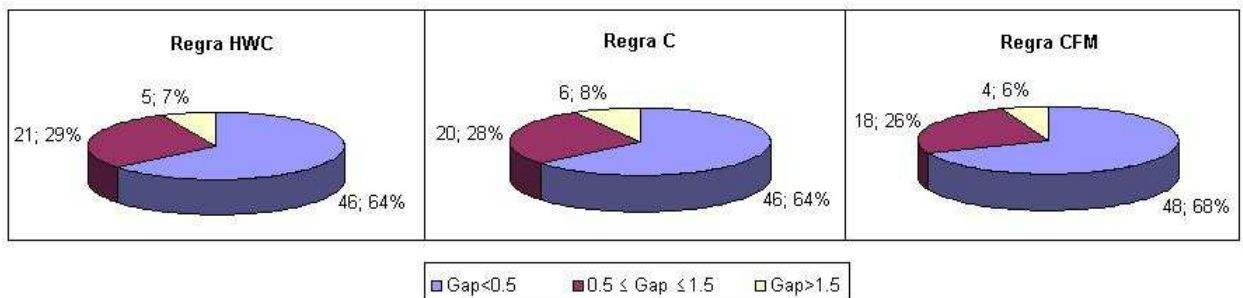


Figura 5.1: Distribuição do *gap* do método subgradiente

Na Figura 5.1 podemos observar três gráficos representativos da distribuição do *gap* e podemos concluir que a maioria dos *gaps* em todas as regras é inferior a 0.5 e que a regra que tem menos *gaps* superiores a 1,5 é a de *Held, Wolf & Crowder*. Contudo

as três direcções apresentam uma distribuição similar do *gap* demonstrando que é um método que obtém resultados com grande qualidade.

5.2 Resultados obtidos pelo método subgradiente radar

Nesta secção são apresentados os resultados computacionais obtidos pelo método subgradiente radar em tabelas cuja primeira coluna, denotada por *Prob* e por $|V|$, identifica o problema e o número de nodos; a segunda coluna indica o valor do parâmetro H considerado; a terceira e quarta coluna, denotadas por *LimSup* que identifica o limite superior (obtido por uma heurística, consultar [15] para detalhes) e *LimLin* que identifica o limite inferior obtido pela relaxação linear HMCF_L ; as duas colunas seguintes referem-se aos resultados obtidos pela relaxação lagrangeana quando usado este método, a primeira destas duas colunas, denotada por *valor* e por *Gap* (em baixo), mostra o melhor limite inferior e o correspondente valor do gap definido por $\text{gap} = \frac{\text{LimSup} - \text{valor}}{\text{LimSup}} \times 100$ e a segunda destas duas colunas, denotada por *Iteração* e por *tempo* (em baixo), mostra o número de iterações usadas pelo procedimento de optimização por subgradiente radar indicado para obter o melhor limite inferior registado na coluna anterior e o tempo computacional, em segundos, usado para obter tal valor.

Notamos que quando o $\text{LimSup} = \text{LimLin}$ este valor corresponde ao valor da solução óptima. Consequentemente, sempre que se fizer referência ao valor óptimo de uma certa instância, este coincide com o valor *LimLin*.

De modo a obtermos os resultados computacionais, começámos por executar o método subgradiente radar para todos os testes já referidos. Os primeiros tamanhos de passo considerados, tal como no método subgradiente, foram 0.5, 1.0, 1.5 e 2.0 com número máximo de iterações para instâncias TC e TR usando 2000 e para intâncias TE usando 5000 iterações máximas.

Os melhores resultados obtidos foram tabelados e identificados todos os casos em que o método subgradiente radar não alcançou *LimLin* (limite inferior da relaxação linear). De seguida, iniciámos o processo de melhoramento destes resultados de forma a

que o valor obtido fosse o melhor possível com o menor número de iterações e usando o menor tempo de execução. Para tal, foi necessário alterarmos sucessivamente os valores do tamanho do passo e do número máximo de iterações, tal como fizemos para o método subgradiente.

Mais uma vez, estas alterações, quer individualmente quer em simultâneo, permitiram melhorar os resultados computacionais visto que conseguimos uma melhoria quer do valor obtido, quer do número de iterações ou do tempo de execução.

Verificámos para que tamanhos de passo se obtinham os melhores resultados e, alterando este valor de tamanho de passo inicial fomos executando novamente e registando os resultados obtidos. Posteriormente, alterávamos o número máximo de iterações, executávamos e registávamos. Por fim, conjugávamos estas duas estratégias de modo a efectuarmos uma análise comparativa e apenas tabelámos os melhores resultados alcançados entre todos.

Os melhores resultados obtidos podem ser observados nas Tabelas 5.5, 5.6 e 5.7. A primeira tabela refere-se aos resultados obtidos para os testes TC, a segunda tabela aos resultados para os testes TE e a terceira tabela aos resultados para os testes TR. Por observação da Tabela 5.5 podemos verificar que para todos os testes com 40 e 60 nodos não foi possível obter o *LimLin*, tal como para TC3_20 ($H = 3$) e TC4_20 ($H = 4$). O *gap* para a grande maioria dos testes é diferente de 0.00, o que pode demonstrar que o valor alcançado pode ser de fraca qualidade. Por observação do número de iterações usadas, podemos concluir que, até para os testes com 20 nodos, é difícil alcançar o melhor valor obtido e necessita de muitas iterações para tal. Apenas para TC2_20 ($H = 4$) e TC5_20 ($H = 5$) é que o número de iterações é inferior a 2000.

A esmagadora maioria dos testes necessita de mais de 4000 iterações para alcançarem um valor aproximado do *LimLin*. Em termos de tempo de execução, o método subgradiente radar não se revela um método rápido em especial para os testes de 60 nodos, que chegam a alcançar os 1312,67 segundos, cujo valor determinado não é muito próximo de *LimLin* e o *gap* evidencia a fraqueza do valor alcançado (TC1_60 ($H = 5$)).

O teste que apresentou os melhores resultados quer pelo valor obtido, quer pelo valor do *gap*, quer pelo número de iterações e tempo computacional usado foi o teste

Tabela 5.5: Resultados computacionais para os testes TC.

$Prob, V $	H	$LimSup$	$LimLin$	$Valor$ (Gap)	$Iteração$ ($Tempo$)
TC1, 20	3	340	339	339 (0.29)	6139 (15.86)
	4	318	318	318 (0.00)	8361 (31.11)
	5	312	312	312 (0.00)	5332 (25.23)
TC2, 20	3	365	364	364 (0.27)	7655 (19.28)
	4	338	338	338 (0.00)	1838 (6.64)
	5	332	332	332 (0.00)	4457 (20.97)
TC3, 20	3	343	339	338 (1.46)	4160 (10.37)
	4	306	306	306 (0.00)	6457 (23.86)
	5	296	296	296 (0.00)	8965 (42.38)
TC4, 20	3	390	387	387 (0.77)	2703 (6.92)
	4	376	376	375 (0.27)	6969 (24.44)
	5	364	364	364 (0.00)	6366 (30.74)
TC5, 20	3	347	347	347 (0.00)	5635 (14.09)
	4	326	324	324 (0.61)	6810 (24.25)
	5	310	310	310 (0.00)	1 (0.00)
TC1, 40	3	609	605	600 (1.48)	6206 (84.55)
	4	548	547	543 (0.91)	8681 (214.53)
	5	522	522	513 (1.72)	6424 (264.69)
TC2, 40	3	566	566	562 (0.71)	2225 (30.03)
	4	519	518	511 (1.54)	7056 (171.52)
	5	496	494	487 (1.81)	6828 (285.83)
TC1, 60	3	866	862	854 (1.39)	4481 (243.72)
	4	781	778	764 (2.18)	5864 (586.84)
	5	734	734	723 (1.50)	7846 (1312.67)

TC5_20 ($H = 5$). Se para as instâncias TC os resultados alcançados já não foram os que esperávamos, para as instâncias TE os resultados foram ainda muito piores do que esperávamos porque o $LimLin$ só foi alcançado para TE2_20 ($H = 3$), TE3_20 ($H = 3$)

Tabela 5.6: Resultados computacionais para os testes TE.

$Prob, V $	H	$LimSup$	$LimLin$	$Valor$ (Gap)	$Iteração$ ($Tempo$)
TE1, 20	3	449	444	438 (2.45)	7371 (20.14)
	4	385	385	379 (1.56)	7371 (30.60)
	5	366	364	356 (2.73)	8531 (45.17)
TE2, 20	3	435	435	435 (0.00)	6659 (18.17)
	4	404	401	395 (0.00)	8415 (18.17)
	5	383	381	372 (2.87)	8656 (45.72)
TE3, 20	3	435	435	435 (0.00)	4764 (13.08)
	4	396	395	392 (1.01)	8498 (33.81)
	5	372	372	368 (1.08)	7704 (39.49)
TE4, 20	3	448	444	440 (1.79)	6015 (16.54)
	4	402	400	393 (2.24)	6676 (26.63)
	5	382	377	369 (3.40)	7534 (39.83)
TE5, 20	3	428	428	426 (0.47)	6012 (16.37)
	4	376	376	376 (0.00)	7744 (29.97)
	5	354	354	353 (0.28)	6304 (32.34)
TE1, 40	3	708	702	690 (2.54)	7265 (113.74)
	4	627	625	607 (3.19)	8268 (233.97)
	5	590	588	565 (4.24)	5592 (258.44)
TE2, 40	3	710	701	688 (3.10)	5981 (91.18)
	4	625	620	600 (4.00)	8831 (244.50)
	5	581	579	558 (3.96)	7915 (352.53)
TE1, 60	3	1525	1503	1482 (2.82)	8689 (531.50)
	4	1345	1315	1261 (6.24)	8897 (991.07)
	5	1238	1213	1151 (7.03)	8064 (1558.64)

e TE5_20 ($H = 4$).

O número de iterações usadas para cada teste TE ultrapassa as 4000 iterações, como é o caso do teste TE1_60 ($H = 4$) que necessitou de 8897 iterações para alcançar um

valor com 46 unidades de diferença do *LimLin*.

O valor do *gap* é elevado para a maioria dos testes, o que revela que os resultados obtidos são de fraca qualidade. Além disso, o tempo de execução é elevado pelo que podemos concluir que as instâncias TE são difíceis independentemente do número de nodos. Por observação da Tabela 5.7, os testes para os quais o método se mostrou menos eficaz, foram essencialmente os testes com $H = 4$ e $H = 5$.

Para as instâncias TR, o método também necessita de muitas iterações para alcançar o *LimLin*, excepto para TR1_20 ($H = 5$) e TR3_20 ($H = 5$). Também o tempo de execução continua elevado mas de um modo geral o método subgradiente radar, para estas instâncias, é mais fácil do que para as restantes.

O número máximo de iterações foi de 7662 para TR4_20 ($H = 4$) com um tempo de execução relativamente rápido comparativamente aos testes TR1_60 como se pode observar na Tabela 5.7. O tempo máximo de execução foi de 925,87 segundos para TR1_60 ($H = 5$).

Quando observamos os resultados obtidos pelo método subgradiente radar podemos verificar que só para as instâncias TC e TR é que foi possível alcançar o *LimLin* para a maioria dos testes. Os tempos de execução para 20 nodos variam entre 0.00 e 45.72 segundos, para 40 nodos variam entre 30.03 e 352.53 segundos. Para 60 nodos, os tempos variam entre 243.72 e 1558.64 segundos.

Na Figura 5.2 podemos observar como o *gap* se distribui para o método subgradiente radar e podemos também verificar que os valores alcançados não são de grande qualidade, visto que 35% desses valores têm *gap* superior a 1.5.

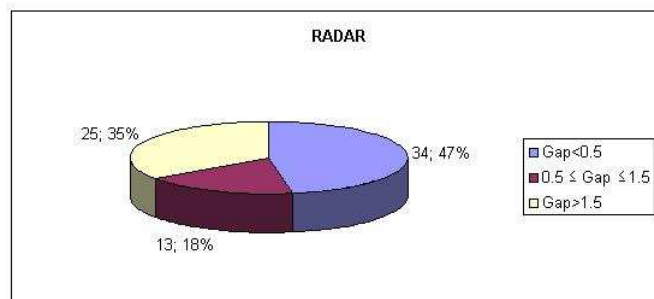


Figura 5.2: Distribuição do *gap* do método subgradiente radar

Tabela 5.7: Resultados computacionais para os testes TR.

$Prob, V $	H	$LimSup$	$LimLin$	$Valor$ (Gap)	$Iteração$ ($Tempo$)
TR1, 20	3	168	168	168 (0.00)	2829 (7.87)
	4	146	146	146 (0.00)	2829 (13.05)
	5	137	137	137 (0.00)	1 (0.00)
TR2, 20	3	201	201	201 (0.00)	2979 (7.81)
	4	161	161	161 (0.00)	5949 (22.96)
	5	140	140	140 (0.00)	3825 (19.49)
TR3, 20	3	157	153	153 (2.55)	6691 (17.40)
	4	130	130	130 (0.00)	1106 (4.40)
	5	121	121	121 (0.00)	1 (0.00)
TR4, 20	3	183	183	183 (0.00)	3858 (10.43)
	4	158	158	158 (0.00)	7662 (29.82)
	5	142	142	142 (0.00)	4694 (23.29)
TR5, 20	3	148	148	148 (0.00)	6106 (16.28)
	4	121	121	118 (2.48)	6902 (27.60)
	5	110	110	106 (3.64)	5902 (30.44)
TR1, 40	3	176	176	176 (0.00)	4704 (68.95)
	4	149	148	147 (1.34)	4968 (133.88)
	5	139	139	138 (0.72)	4971 (230.36)
TR2, 40	3	219	219	219 (0.00)	5963 (85.86)
	4	176	176	176 (0.00)	6628 (179.59)
	5	155	155	154 (0.65)	6783 (308.68)
TR1, 60	3	274	274	274 (0.00)	6264 (405.22)
	4	207	207	206 (0.48)	4165 (478.66)
	5	189	188	186 (1.59)	4947 (925.87)

A partir da observação das tabelas apresentadas nesta secção podemos constatar que este método comparativamente ao anterior demonstrou-se muito menos eficaz visto que apresenta tempos de execução muito mais elevados, necessita de muitas iterações

5.3. COMPARAÇÃO DOS RESULTADOS OBTIDOS ENTRE OS DOIS MÉTODOS⁶⁵
para alcançar valores que são piores que os obtidos pelo método subgradiente.

5.3 Comparação dos resultados obtidos entre os dois métodos

Nesta secção fazemos uma comparação dos resultados computacionais obtidos pelo método subgradiente e pelo método subgradiente radar para a relaxação lagrangeana do problema HMCF descrita no Capítulo 4.

Observando a Figura 5.3, podemos comparar a distribuição do *gap* entre os dois métodos, podemos concluir que, tal como já tínhamos referido, a distribuição do *gap* no método subgradiente para as três direcções é semelhante mas quando comparamos com o método subgradiente radar torna-se evidente a diferença de comportamento entre os métodos. Ao contrário do método subgradiente, em que o número de testes cujo valor não melhora vai diminuindo nos intervalos de referência do *gap*, no método subgradiente radar o número de testes cujo valor não melhora no intervalo de referência do *gap* com valor superior a 1.5 aumenta. Ou seja, quando o *gap* é menor ou igual a 1.5 é o método subgradiente radar que tem o menor número de testes a obter estes valores, evidenciando-se um método cuja qualidade dos valores alcançados é mais fraca em relação ao método subgradiente.

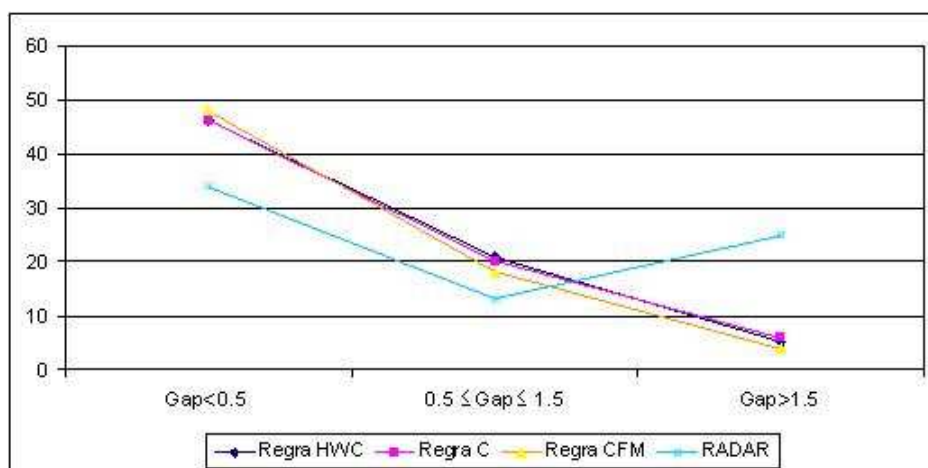


Figura 5.3: Distribuição do *gap* dos dois métodos

Fazendo uma análise comparativa entre as instâncias de cada um dos métodos e o *gap* para 20 nodos resulta a Figura 5.4. Observando esta figura, podemos concluir que para a maioria das instâncias o *gap* é inferior a 0.5, excepto para a instância TE de *Held, Wolf & Crowder* e de *Radar*. Para *Held, Wolf & Crowder*, o número de *gaps* é igual para os intervalos do *gap* menores que 0.5 e para intervalos do *gap* maiores ou iguais a 0.5 e menores ou iguais a 1.5. Para o método *Radar* o número de *gaps* superior a 1.5 supera os inferiores a 0.5 para as instâncias TE.

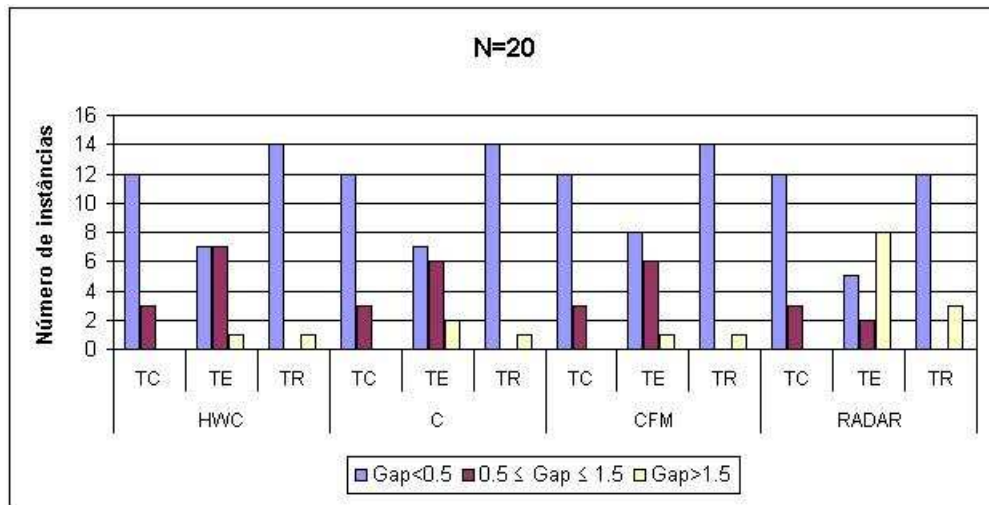


Figura 5.4: Distribuição do *gap* dos testes com 20 nodos

As conclusões diferem quando fazemos uma análise análoga para 40 nodos como podemos observar na Figura 5.5, uma vez que as instâncias TE se destacam das outras pelo facto do *gap* para o método subgradiente ser menor ou igual a 1.5 para as três regras enquanto que, para o método subgradiente radar o *gap* para TC e TE tem muitos testes com valor acima de 1.5. Mas, de um modo geral, as instâncias TE são as que se revelam mais difíceis para qualquer método usado, cujo *gap* para ambos os métodos é sempre superior a 0.5.

Para 60 nodos podemos concluir de imediato, por observação da Figura 5.6, que os testes TE são os que têm pior qualidade visto que o valor do *gap* para todos os testes é superior a 1.5 para qualquer regra ou método usado. Os testes TR com 60 nodos são os que revelam mais qualidade, uma vez que muitos deles têm valor do *gap* inferior a 0.5.

5.3. COMPARAÇÃO DOS RESULTADOS OBTIDOS ENTRE OS DOIS MÉTODOS⁶⁷

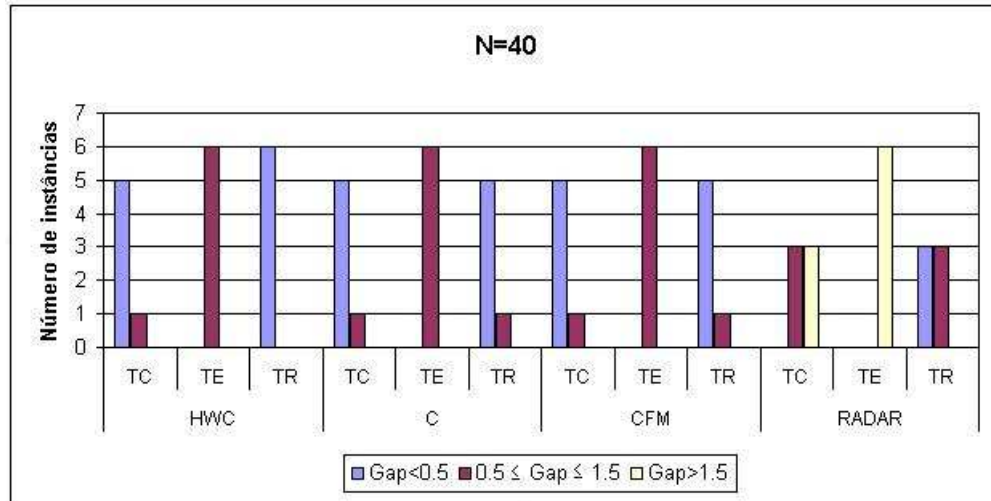


Figura 5.5: Distribuição do *gap* dos testes com 40 nodos

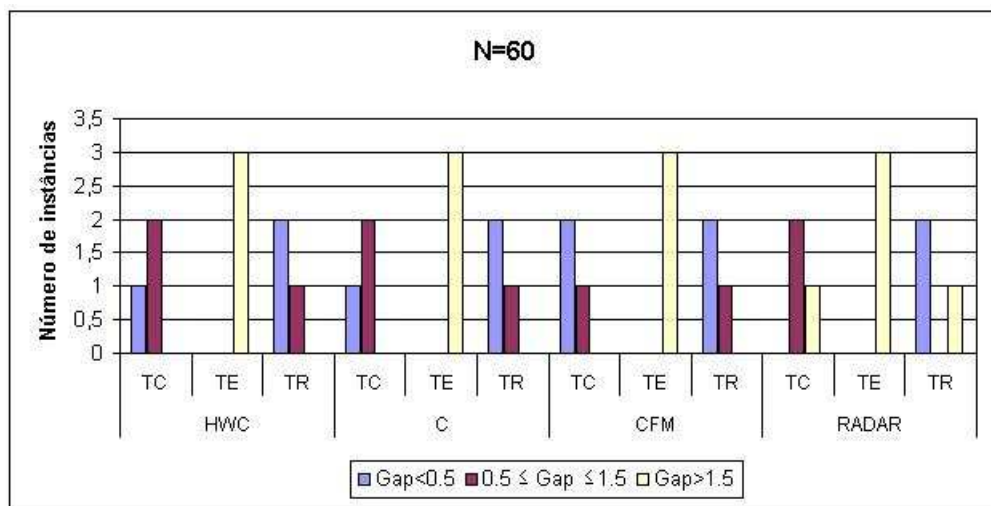


Figura 5.6: Distribuição do *gap* dos testes com 60 nodos

De um modo geral, podemos concluir que para 20 nodos em todos os métodos usados o número de testes com *gap* inferior a 0.5 é o que acontece com mais frequência. No entanto, para 40 nodos já não podemos dizer o mesmo do método subgradiente radar uma vez que para este método, o número de *gaps* que acontece com mais frequência é para valores superiores a 1.5. Para 60 nodos, os valores do *gap* superiores a 1.5 são os que aparecem com mais frequência em ambos os métodos como podemos observar na Figura 5.6.

Fazendo agora uma análise à distribuição do *gap* pelos testes podemos observar na Figura 5.7 que para as regras *HWC*, *C* e *CFM* com 20 e 40 nodos, a maioria dos testes têm *gaps* inferiores a 0.5. Para 60 nodos as regras *HWC* e *C* têm um valor idêntico para a distribuição do *gap*, a regra de *CFM* é a que revela mais testes com *gap* inferior a 0.5. No entanto, pelo método subgradiente radar para 40 e 60 nodos a quantidade de testes que têm *gap* superior a 1.5 são os mais frequentes e mais uma vez podemos concluir que o método subgradiente radar é o que demonstra menor qualidade dos valores alcançados.

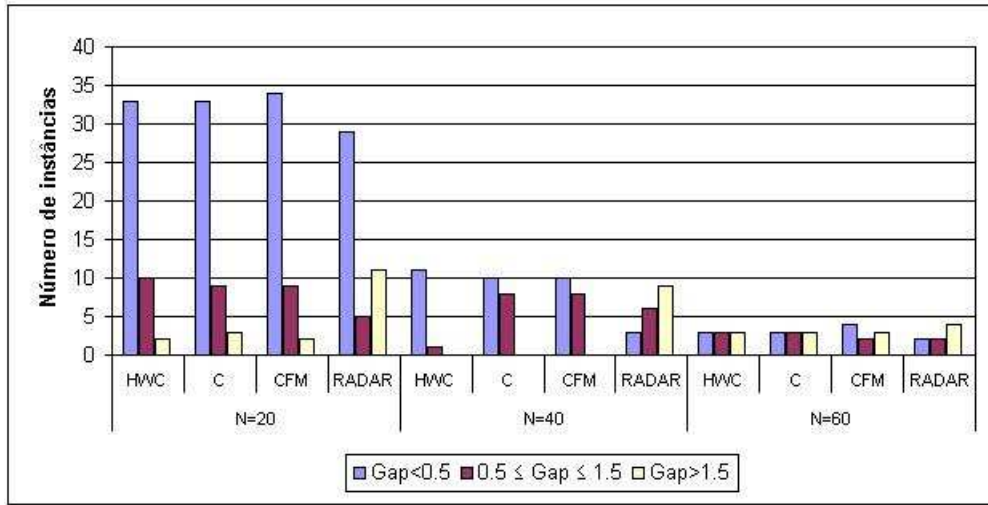


Figura 5.7: Distribuição do *gap* pelos métodos para testes com 20, 40 e 60 nodos

Para qualquer valor do parâmetro H (3, 4 ou 5) o número de *gaps* é maioritariamente inferior a 0.5 para todos os métodos (Figura 5.8). No entanto, não descurar que para $H=5$ no método subgradiente radar o comportamento do *gap* é distinto, como já pudémos observar anteriormente, visto que o número de testes com *gap* inferior a 0.5 é similar ao número de testes com *gap* superior a 1.5. Os valores intermédios acontecem menos vezes e este tipo de comportamento não se verifica nas direcções do método subgradiente.

Nas Figuras 5.9 e 5.10, podemos observar como se comporta cada método em termos de tempo para $N=20$, $N=40$ e $N=60$ ou para $H=3$, $H=4$ ou $H=5$, respectivamente.

Para 20 nodos podemos verificar que os tempos são bastante reduzidos, apenas o método subgradiente radar se aproxima dos 50 segundos. Para 40 nodos, os tempos

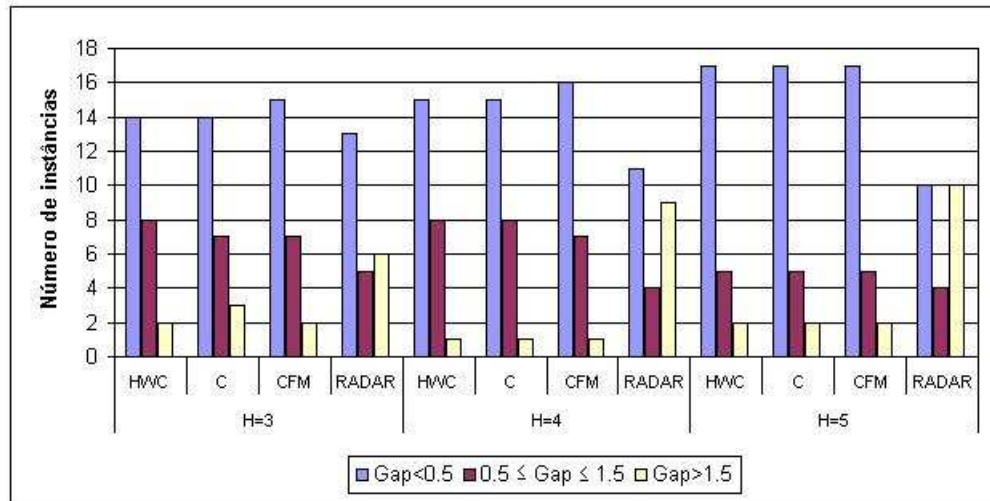


Figura 5.8: Distribuição do *gap* para o parâmetro *H*

já são mais variados contudo o método subgradiente radar é o que apresenta tempos de execução mais elevados e mais uma vez para este método, é o único que apresenta todos os testes de 60 nodos com tempos superiores a 200 segundos.

Observando os tempos para o parâmetro *H* podemos encontrar uma grande variedade de tempos em ambos os testes, independentemente do método usado.

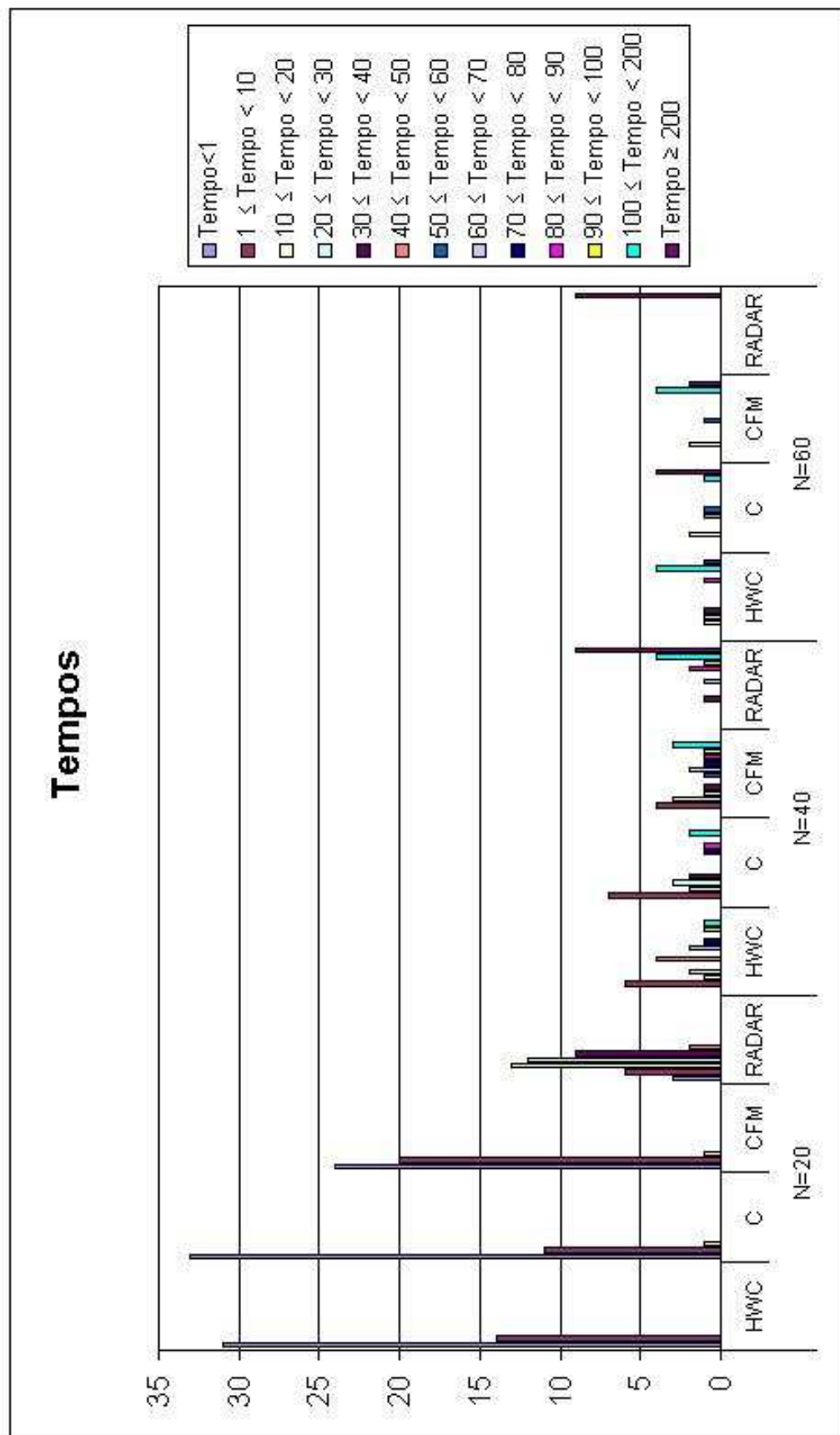


Figura 5.9: Distribuição dos tempos para 20, 40 ou 60 nodos

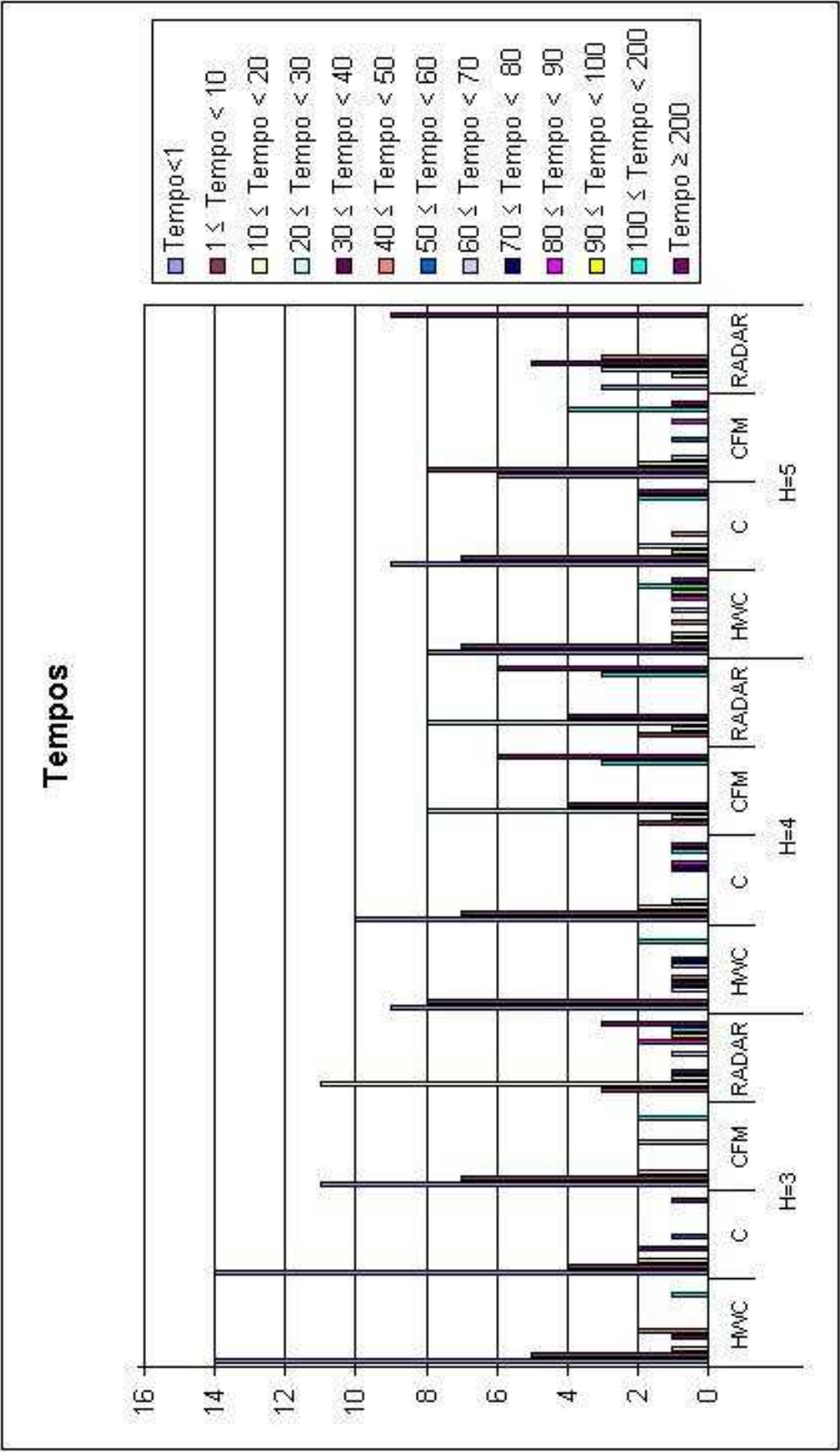


Figura 5.10: Distribuição dos tempos para o parâmetro H

Capítulo 6

Conclusão

Neste trabalho apresentamos a técnica da relaxação lagrangeana, métodos de otimização para a sua resolução e um problema de Otimização Combinatória que resolvemos recorrendo a uma sua relaxação lagrangeana. Descrevemos três métodos de optimização bastante conhecidos que usam subgradientes na resolução do problema dual lagrangeano: o Método Subgradiente, o Método Subgradiente Radar e o Método do Feixe de Subgradientes.

Elaborámos uma breve descrição destes três métodos e o modo de implementá-los para a resolução do nosso problema. Procedemos à sua implementação mas devido à complexidade do método do feixe de subgradientes não foi possível obter resultados computacionais para o problema HMST com este método.

No método subgradiente foram utilizadas três formas de determinar a direcção: regra de *Held, Wolf & Crowder*, regra de *Crowder* e regra de *Camerini, Fratta, Maffioli modificada*; sendo a de *Crowder* a mais promissora e a de *Camerini, Fratta, Maffioli modificada* a menos, apesar das três direcções apresentarem um comportamento semelhante.

Comparando, usando os resultados computacionais obtidos para os dois métodos podemos concluir que o método subgradiente radar é menos eficiente na obtenção do *LimLin*, usando um maior número de iterações e mais tempo de execução, especialmente para os testes com 40 e 60 nodos.

Apesar de, na literatura, o método subgradiente radar se revelar um método bastante

eficaz e melhor que o método subgradiente, essa não foi a nossa experiência. No entanto, salvaguardamos a possibilidade de os valores alcançados poderem vir a ser melhorados, visto que a escolha dos dados iniciais é um factor muito importante se não mesmo um factor decisivo para a obtenção de bons resultados.

Em qualquer um dos métodos analisados, a escolha dos dados iniciais bem como os parâmetros escolhidos para melhorarem a eficácia dos métodos, é muito importante visto serem estes parâmetros decisivos na forma de encontrar a melhor estratégia para alcançar o melhor valor possível na resolução dos problemas e do modo mais eficiente.

De um modo geral, podemos concluir que o método subgradiente continua a ser um método bastante eficaz e que ao longo do nossa tese se revelou o melhor. No entanto, novas estratégias podem ser desenvolvidas para que ainda se consiga mais eficácia e mais rapidez na resolução de problemas cada vez mais complexos.

Bibliografia

- [1] C. Beltran, F.J. Heredia. An Effective Line Search for the Subgradient Method. *Journal of Optimization Theory and Applications*, 125: 1–18, 2005.
- [2] C. Beltran-Royo. *Generalized unit commitment by the radar multiplier method*. Dept. d'Estadística i Investigació Operativa, Universitat Politècnica de Catalunya, 2001, Ph.D. Thesis.
- [3] C. Beltran-Royo. The radar method: An effective line search for piecewise linear concave functions. A ser publicado em "Annals of Operations Research", 2008.
- [4] J.F. Bonnans, J.C. Gilbert, C. Lemaréchal and C.A. Sagastizábal. *Numerical Optimization*. Springer-Verlag Berlin Heidelberg, 2006.
- [5] P.M. Camerini, L. Fratta, F. Maffioli. On improving relaxation methods by modified gradient techniques. *Mathematical Programming Study* 3 (1975) 26-34.
- [6] T. Crainic, A. Frangioni and B. Gendron. Bundle-based relaxation methods for multicommodity capacitated fixed charge network design. *Discrete Applied Mathematics*, 112: 73–99, 2001.
- [7] H. Crowder. Computational Improvements for Subgradient Optimization. *Symposia Mathematica*, Vol XIX, Academic Press, London (1976).
- [8] G. Dahl, L. Gouveia and C. Requejo. On formulations and methods for the hop-constrained minimum spanning tree problem. In *Handbook of Optimization in Telecommunications*, edited by M.G.C. Resende and P.M. Pardalos, Springer Science + Business Media, 2006.

- [9] M.L. Fisher. An applications oriented guide to Lagrangean relaxation. *Interfaces*, 15(2):10–21, 1985.
- [10] M. Fischetti, P. Toth. An efficient algorithm for the min-sum arborescence problem on complete digraphs. *ORSA Journal on Computing*, 5, 426–434, 1993.
- [11] A. Frangioni. *Dual-ascent methods, and multicommodity flow problems*. Dipartimento di informatica, Università di Pisa, 1997, Ph.D. Thesis.
- [12] A.M. Geoffrion. Lagrangean relaxation for integer programming. *Mathematical Programming Study*, 2:82–114, 1974.
- [13] L. Gouveia. Multicommodity Flow Models for Spanning Trees with Hop Constraints. *European Journal of Operational Research*, 95:178-190, 1996.
- [14] L. Gouveia. Using Variables Redefenition for Computing Lower Bounds for Minimum Spanning and Steiner Trees with Hop Constraints. *INFORMS Journal on Computing*, 10(2):180-188,1998.
- [15] L. Gouveia and C. Requejo. A new lagrangean relaxation approach for the hop-constrained minimum spanning tree problem. *European Journal of Operational Research*, 132(3):539–552, 2001.
- [16] M. Held, P. Wolfe, and H.P. Crowder. Validation of subgradient optimization. *Mathematical Programming*, 6:62–88, 1974.
- [17] T. Magnanti and L. Wolsey. Optimal Trees. In M.O. Ball, T.L. Magnanti, C.L. Monma, and G.L. Nemhauser, editors, *Network Models, Handbooks in Operations Research and Management Science, Vol. 7*. Elsevier Science Publishers, North-Holland, 1995.
- [18] E. Monteiro e F. Boavida. *Engenharia de Redes Informáticas*. FCA - Editora de Informática, 2000.
- [19] G.L. Nemhauser and L.A. Wolsey. *Integer and Combinatorial Optimization*. John Wiley & Sons, 1988.

- [20] B.T. Polyak. Minimization of nonsmooth functionals. *USSR Computational Mathematical and Mathematical Physics*, 9 (1969) 14-29.
- [21] N.Z. Shor. *Minimization methods for non-differentiable functions*. Springer Verlag, Berlin, 1985.